3D CAPTCHA

A Next Generation of the CAPTCHA

Montree Imsamai and Suphakant Phimoltares*
Advanced Virtual and Intelligent Computing (AVIC) Research Center
Department of Mathematics, Faculty of Science, Chulalongkorn University
Patumwan, Bangkok, Thailand
montree.i@chula.ac.th, suphakant.p@chula.ac.th

***Abstract*—Nowadays, the Internet is now becoming a part of our everyday lives. Many services, including Email, search engine, and web board on Internet, are provided with free of charge and unintentionally turns them into vulnerability services. Many software robots or, in short term, bots are developed with purpose to use such services illegally and automatically.**

Thus, web sites employ human authentication mechanism called Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) to counter this attack. Unfortunately, many CAPTCHA have been already broken by bots and some CAPTCHA are difficult to read by human.

In this paper, a new CAPTCHA method called 3D CAPTCHA is proposed to provide an enhanced protection from bots. This method based on assumption that human can recognize 3D character image better than Optical Character Recognition (OCR) software bots.

Keywords- CAPTCHA; 3D; Optical Character Recognition; bot

I. INTRODUCTION

The Turing Test [1] is a test for determining whether or not machine intelligence can converse like a human. This test has three participants – two subjects and judge. One of the subjects is a person and the other is a computer. Both subjects are hidden from the view of the judge. They communicate with the judge via text-only channels. The role of the judge is to determine which text channel corresponds to the human and which corresponds to the computer. If the judge cannot determine this, then the computer passes the test.

Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) is similar to Turing Test but the judge is a computer [2]. The basic idea is that the distorted image of a sequence of letters and numbers is shown to a user and asked to type that sequence. The user will have no problem to recognize all letters and numbers, but the computer programs or bots cannot recognize them.

The CAPTCHA is usually a simple visual test or puzzle that a human can complete without much difficulty, but an automated program cannot understand. The test usually consists of letters, numbers or their combination with overlapping and intersection. The CAPTCHA images may be distorted in some way or shown against an intricate background to keep them from being easily read by Optical Character Recognition (OCR) software.

The CAPTCHA is used to keep bots and other automated programs from signing up for offers, collecting or signing up for email address, violating privacy, trying to crack passwords, or sending out spam to unsuspecting email recipients.

The CAPTCHA methods can be divided into two groups: OCR-based and non-OCR-based methods as follows:

1- OCR-based method: The distorted image of a word is shown to the user. After that the user is asked to type that word. This method is based on the weakness of the OCR software because this software has difficulty reading text from distorted image. Examples of these methods are used by Google, Hotmail, and Yahoo. In this paper, a new type of OCR-based method is proposed. The detail of the existing OCR-based methods is provided in the related works section.

2- Non-OCR-based method: Instead of show the distorted image of a word and ask user to type it. This method based on the features of multimedia systems such as pictures, sound, and videos. Examples of these methods are Collage CAPTCHA [3], Text-to-Speech CAPTCHA [4], and The CAPTCHA method that found on Spamfizzle [5] and CNET [6] web site. In our recent survey, we found that the CAPTCHA from Spamfizzle and CNET can be categorized into 3D form of different way with respect to our proposed method. For Spamfizzle, the CAPTCHA is a 3D model image shown to users and allowed them to type the answer of that image. For CNET, the CAPTCHA is based on user ability to identify objects in 3D instead of using alphanumeric characters. Both 3D techniques are different to our proposed method because these are not complete 3D CAPTCHA. The CAPTCHA is only 2D image that are prior prepared by the projection method.

To break the CAPTCHA, the simple systems have been cracked through various means, ranging from smart image scanning and analysis software to brute-force guessing, mapping the CAPTCHA image file name with a solution, show the CAPTCHA in another web site and use innocent people to solve it.

The rest of the paper is organized as follows: In Section 2, related works in the real world CAPTCHA applications are investigated. In Section 3, our proposed method is described. The result of our suggested method is illustrated and discussed

* Corresponding author

978-1-4244-5943-8/10/$26.00 ©2010 IEEE

in Section 4 and Section 5, respectively. Section 6 is the conclusion of the paper.

II. Related Works

The following are ten examples of the CAPTCHA methods currently being used on the Internet today:

A. Steam Users' Forums

Steam Users' Forums [7] use the CAPTCHA when anonymous users want to perform a search. This CAPTCHA method is consists of 6 characters including letter and number. The color of character is very dim and background is noisy. A related example is depicted in Fig. 1.

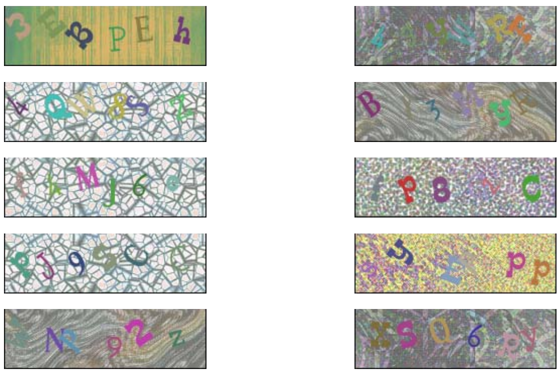

Figure 1. The CAPTCHA used by Steam.

B. Google

Google [8] uses the CAPTCHA in many places. It is shown when URLs are added to Google and a new Gmail account is signed up. It is obvious that Google does not add background to the image. The sequence is shown in a pure white background. The characters are in single color such as green, red, and blue. It can also be seen that a Google CAPTCHA uses many fonts but all the characters in a sequence will be of the same font. The examples are shown in Fig. 2.

Figure 2. The CAPTCHA used by Google.

C. Hotmail

The CAPTCHA method that used by Microsoft [9] is typically present users with a sequence of characters that can be recognized by humans, but not by OCR software. Microsoft's system is used for services including Hotmail, MSN and Windows Live as shown in Fig. 3.

Figure 3. The CAPTCHA used by Hotmail.

D. Friendster

The CAPTCHA used by Friendster [10] use many fonts in a sequence. The character is colorful and easy to read. It is not overlapping between each other. Some examples are shown in Fig. 4

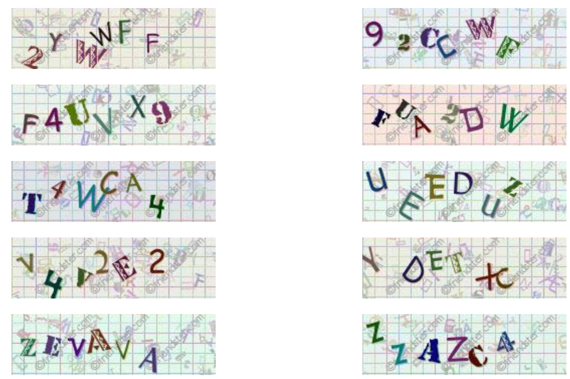

Figure 4. The CAPTCHA used by Friendster.

E. eBay

eBay [11] use the CAPTCHA when a new user want to register for a new account. This method use only number in the sequence and no background is added. Some examples are shown in Fig. 5.

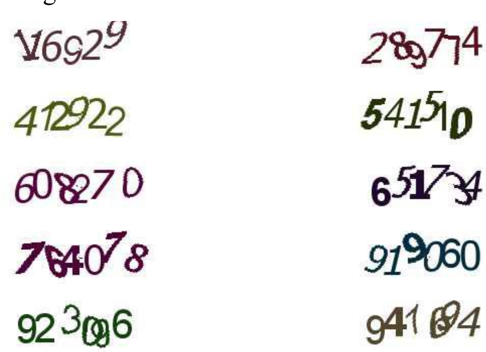

Figure 5. The CAPTCHA used by eBay.

F. Yahoo

The CAPTCHA used by Yahoo [12] is a black character in white background. The length of the sequence is not fix, it random between 6 – 8 characters. Examples of this method are shown in Fig. 6.

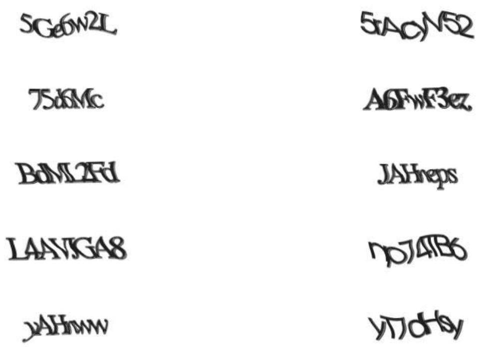

Figure 6. The CAPTCHA used by Yahoo.

G. *Rediffmail*

Rediffmail [13] is free e-mail service web site. Some characters are overlapping but difficult to read. Background is very noisy as shows in Fig. 7.

Figure 7. The CAPTCHA used by Rediffmail.

H. *pc1news*

The CAPTCHA used by pc1news [14] is easy to read. It uses a constant font. No overlapping between each character. Line noise is easy to remove and segmentation is done very easy. Examples are shown in Fig. 8.

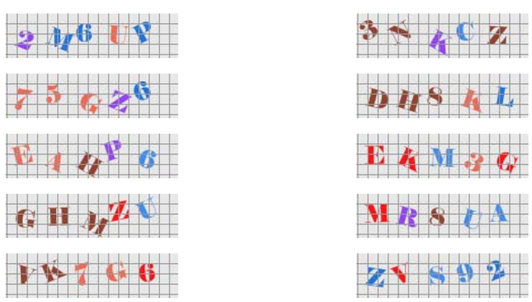

Figure 8. The CAPTCHA used by pc1news.

I. *Jeans*

Jeans Mobile Operator [15] uses this CAPTCHA before users perform to send a SMS. This method is not secure. Due to color model separating to characters is very easy in segmentation. Examples of this method are shown is Fig. 9.

Figure 9. The CAPTCHA used by Jeans.

J. *PHP-CAPTCHA-Class*

This is a free CAPTCHA class [16] that use in PHP web site. This class will generate a form with a CAPTCHA picture with 5 characters long including letters and numbers. It use various font faces. Background texture can confuse user because it include character-like image. Examples are shown in Fig. 10.

Figure 10. PHP-CAPTCHA-Class.

III. OUR PROPOSED METHOD

In this paper, the proposed method has been developed to distinguish human users and computer programs from each other by the same fact that human user can recognize a sequence of 3D characters easily [17] but it makes OCR program a lot of confusing while identifying the characters. There are two major issues involved in building a strong CAPTCHA solution. First, the basis for the puzzle or challenge must be something that is truly difficult for computers to solve. Second, the way puzzles and responses are processed must easy for human users.

The puzzle must be very difficult for computers to solve and relatively easy for humans. Simple character recognition is not one of those problems, despite the fact that users are so used to see it on the sites frequently. On those sites that are successfully using warped text, the real problem preventing scripting is segmentation.

OCR is still not as good as humans to determine where one character ends and the next one begin. The basis for a strong OCR-based CAPTCHA is ensuring that segmentation is hard.

In fact, once segmentation is solved, computers are much better at recognizing individual characters than users are. This means that characters should have some overlap and any decoy lines with thickness. Type should also run in the same direction as the strokes that compose the letters.

In our method, the CAPTCHA text consists of the alphanumeric characters which are letters and numbers. The text is composed of six characters, and each of which has its own axis and rotation angle. Each character is rotated in a certain angle ranging from -45 degree to 45 degree by using standard randomization function. In our experiment, the CAPTCHA images are generated by using DirectX, a programming library consisting of many functions to manipulate a 3D picture as well as a function that draws a 3D text. A web site for evaluation is written in ASP.NET (C# language). This web site gets a survey from users about this proposed method.

In 3D space, an object contains three dimensions corresponding to three axes which are x-axis, y-axis, and z-axis. This relates to many 3D-factors. In this paper, ten proposed factors are selected to adjust a sequence of characters as follows:

- Rotation
- Overlapping
- Obstacle such as straight line
- Distributed Noise
- Character Color
- Background Color
- Scaling
- Font
- Special Character
- Background Texture

These factors are applied to a sequence of alphanumeric characters and can be categorized to 3D CAPTCHA schemes as follows:

Scheme 1: 3D CAPTCHA with rotation, all characters contain a same color but each has its own rotation angles in three axes. Our method uses a left-handed Cartesian coordinate system. While the values on the x-axis increase from left to right. The values on the y-axis increase from bottom to top and the values on the z-axis increase from back to front. Each character is rotated in an individual angle ranging from -45 degree to 45 degree. The range of rotated angle is limited because some characters seem to be the same if the angle is out of the range. This angle is randomly generated by using standard randomization function. Examples of this CAPTCHA are shown as Fig. 11.

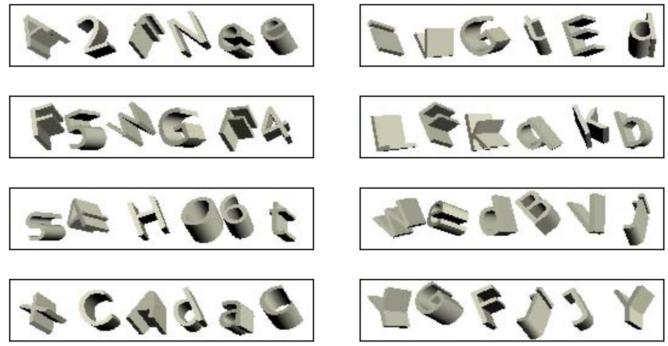

Figure 11. Scheme 1: 3D CAPTCHA with rotation.

Scheme 2: 3D CAPTCHA with overlapping characters. This factor is similar to the first factor but it has some characters overlapping with adjacent characters. Such characters often cause a segmentation problem. Examples of this CAPTCHA are shown as Fig. 12.

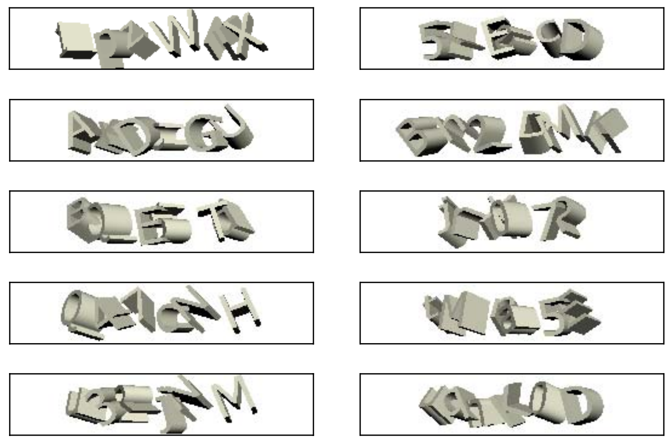

Figure 12. Scheme 2: 3D CAPTCHA with overlapping and rotation.

Scheme 3: 3D CAPTCHA with rotation and straight line is drawn across a sequence of characters. This line is a 3D model and as thick as character. Examples of this CAPTCHA are shown as Fig. 13.

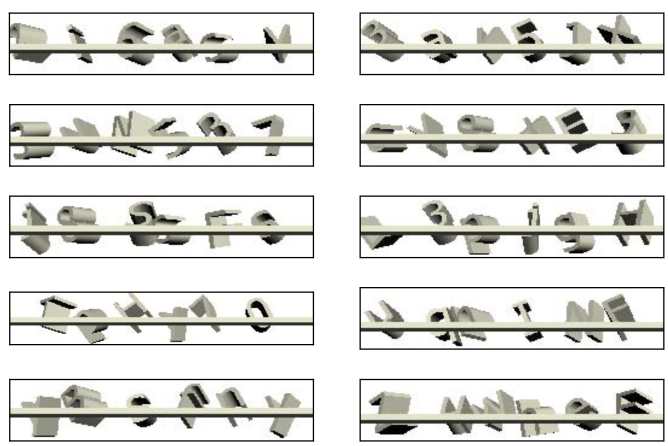

Figure 13. Scheme 3: 3D CAPTCHA with rotation and straight line across a sequence.

Scheme 4: 3D CAPTCHA with distributed noise of characters. This scheme is done by placing noises into the image pixel. It causes the difficulty level is higher of predicting by OCR because of reduction of image clarity. Often OCR software try to remove noise by color filters however, to prevent this, we set noise color as the same color as the character. The number of pixel noises is randomly generated between 20% and 30% with respect to the size of CAPTCHA. Examples of this CAPTCHA are shown as Fig. 14.

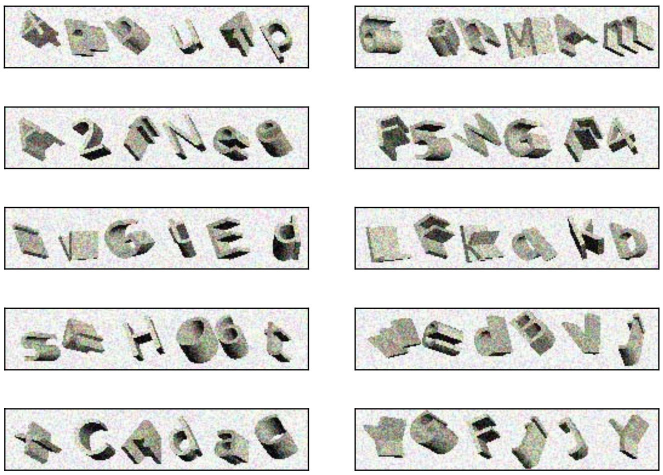

Figure 14. Scheme 4: 3D CAPTCHA with distributed noise and rotation.

Scheme 5: 3D CAPTCHA with rotation and color background. A background color is used as a main factor for this scheme the first step is adding background color. For the next step the color of text in the CAPTCHA is generated randomly, always within a choice of colors that would contrast sufficiently with the background color, to ensure legibility. Examples of this CAPTCHA are shown as Fig. 15.

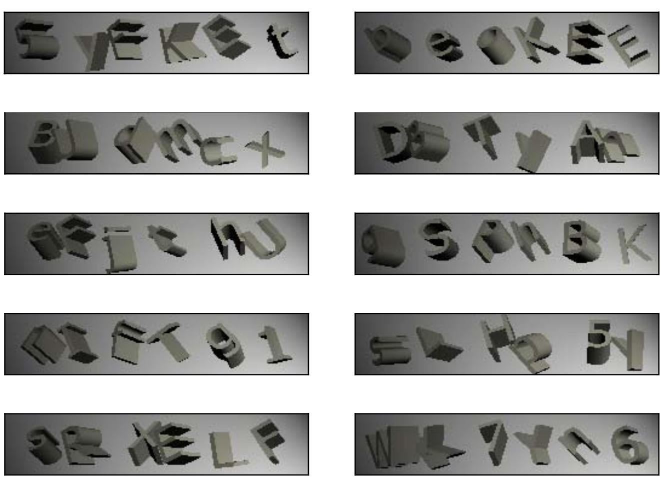

Figure 15. Scheme 5: 3D CAPTCHA with rotation and color background.

Scheme 6: 3D CAPTCHA with rotation and character color variation, each character contains randomized color to attract user and improve usability. Color is extensively used in user interfaces. When used properly, color can much enhance user interface design. Using color has also been common in OCR-based CAPTCHA methods for the following reasons: color is a strong attention-getting mechanism; color can provide variation to fit different user preferences, and color is appealing and can make the CAPTCHA challenges interesting.

In our method, color is randomly generated from red, green and blue color components. Each component represents the intensity of the color in the range of 0 to 255. Examples of this CAPTCHA are shown as Fig. 16.

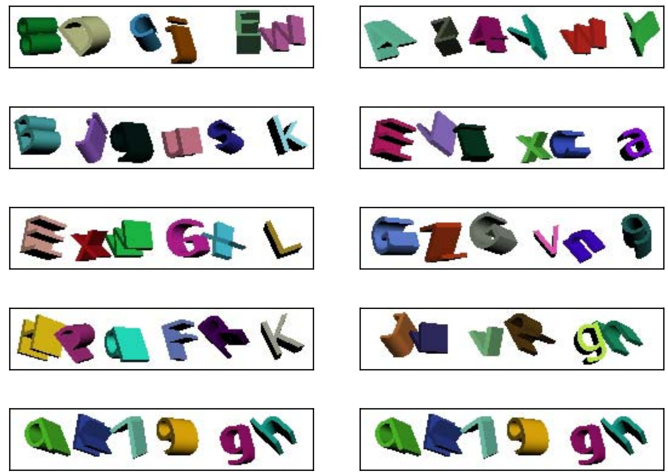

Figure 16. Scheme 6: 3D CAPTCHA with rotation and color variation.

Scheme 7: 3D CAPTCHA with rotation and character scaling, this method not only use rotation but also use scaling in each character. Scaling is performing individual in each axis. Scale value is randomly select to perform a half or double in length. Examples of this CAPTCHA are shown as Fig. 17.

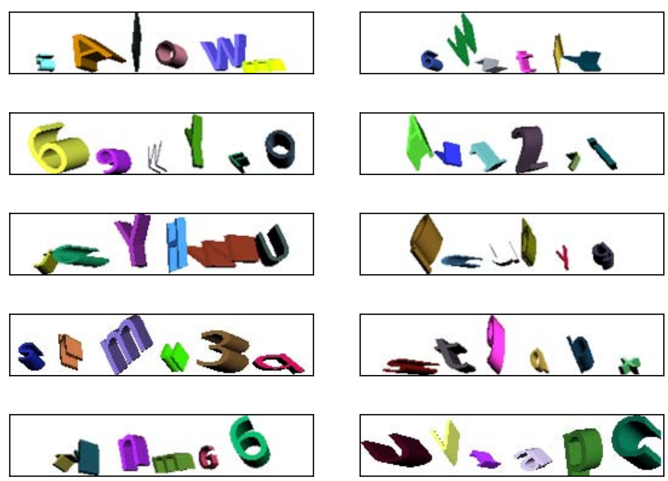

Figure 17. Scheme 7: 3D CAPTCHA with scaling and rotation.

Scheme 8: 3D CAPTCHA with rotation and font variation, font selection can also introduce flaws. Font can be broadly divided into serif and sans-serif fonts. Serif contains the small features at the end of individual characters. Some characters of serif font have unique features for certain characters that can make recognition much easier. For this reason, Sans-serif fonts are recommended to be included in our system. Examples of this CAPTCHA are shown as Fig. 18.

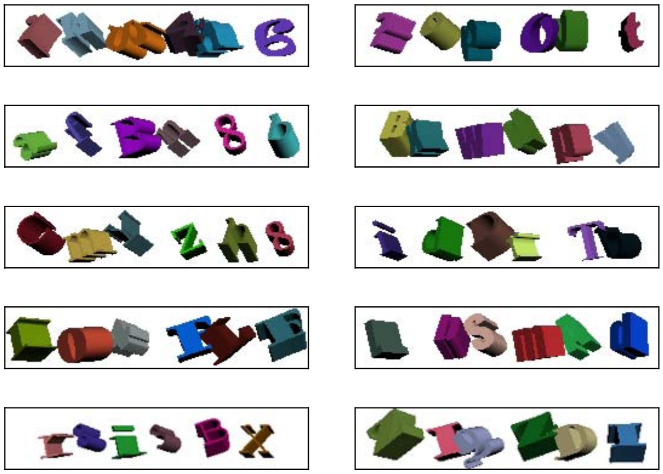

Figure 18. Scheme 8: 3D CAPTCHA with two fonts, overlapping and rotation.

Scheme 9: 3D CAPTCHA with special characters. In order to ensure that the difficulty of OCR is as much as possible, it is important to make sure that there are many possibilities for each character. For instance, if the character set is alphabetic, the recognition task is much simple for software. Use of a full alphanumeric character set is best though typically some characters are eliminated for usability. To increase resistance, it is helpful when the special character is included. Special characters as the following: [!, @, #, $, %, &, *, +, ?] are added to our proposed CAPTCHA. Examples of this CAPTCHA are shown as Fig. 19.

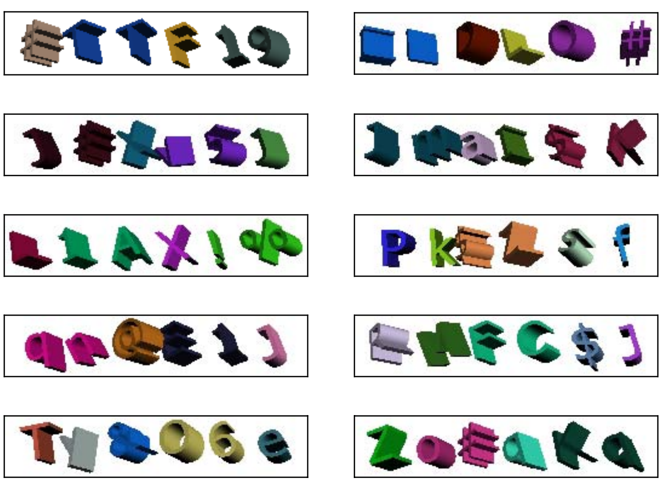

Figure 19. Scheme 9: 3D CAPTCHA with special characters and rotation.

Scheme 10: 3D CAPTCHA with Background Texture. The basic idea is to use texture composed of random primitives to form the background. This scheme is filling a background layer with a random texture. It is more difficult to defeat since it involves two challenging problems in computer vision, namely, image segmentation and texture analysis. It cleverly makes use of the unique capabilities of human visual systems. Examples of this CAPTCHA are shown as Fig. 20.

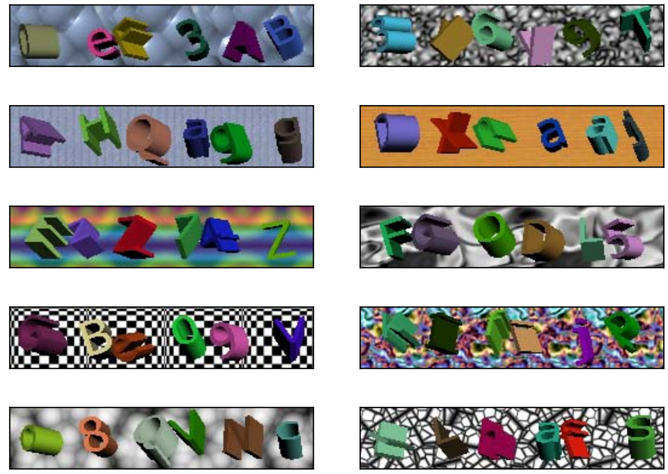

Figure 20. Scheme 10: 3D CAPTCHA with rotation, color variation, and background texture.

IV. RESULTS

An implementation of this proposed method is available at http://aspspider.info/huskar. A screenshot of the evaluation web site is shown as Fig. 21.

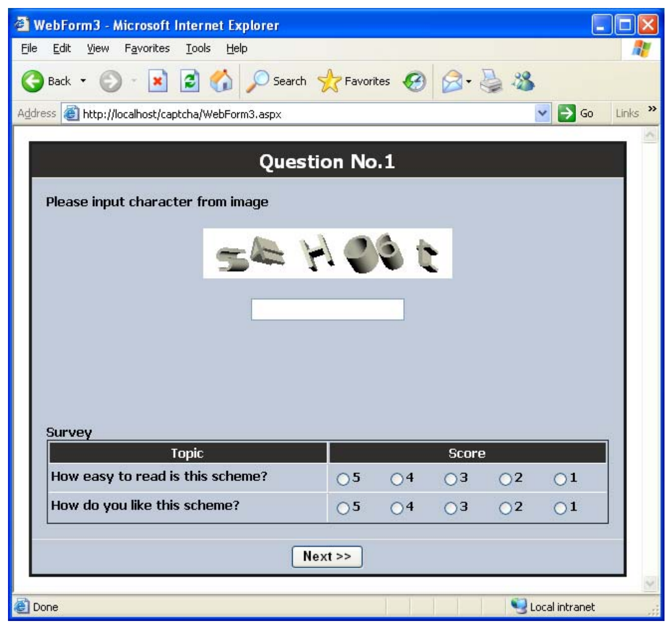

Figure 21. Web Site for Evaluation.

The result of this survey is collected from 138 people with 68 male and 70 female. The survey was comprised of 10 questions. There are ten questions corresponding to ten 3D CAPTCHA methods. For example, question 1 is about 3D CAPTCHA with rotation proposed in the previous section (scheme 1). A survey of usability and preference is also presented in the question.

The web site is used to yield a survey about the readability of our CAPTCHA method. Hit ratio, the percentage of the number of correct recognition by human, is used to measure the efficiency of our 3D CAPTCHA as illustrated in Fig. 22.

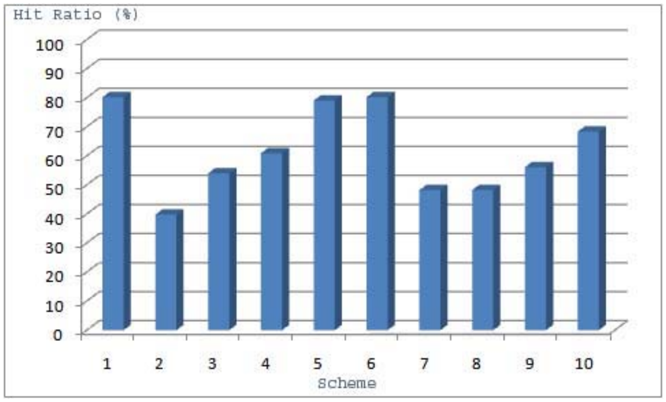

Figure 22. Hit ratio of ten schemes.

The evaluated web site also gets survey about usability and preference of the users. The usability is how easy a string of characters of the considered scheme is to read by human. User can give a score to the CAPTCHA scheme by click a desire score. The score was basically based on 5 points – scale of strongly agree, agree, neutral, disagree and strongly disagree. The result of usability is shown in Fig. 23.

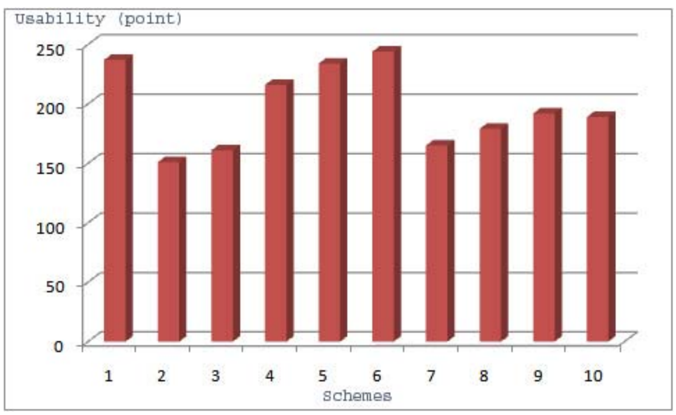

Figure 23. Usability of ten schemes.

The preference is how much user like the considered scheme. The score was also based on 5 points – scale of strongly agree, agree, neutral, disagree and strongly disagree. The result of preference is shown in Fig. 24.

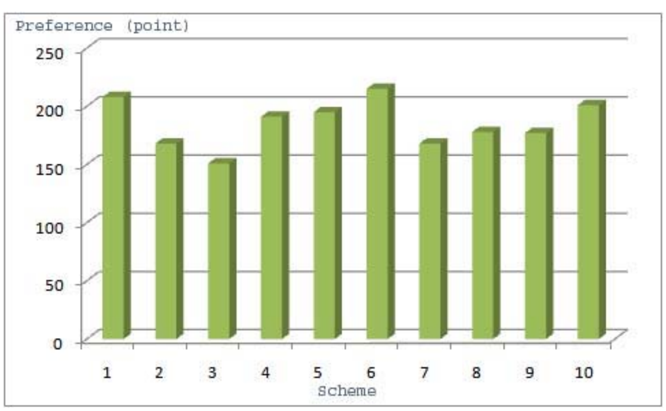

Figure 24. Preference of ten schemes.

From the result of the hit ratio, usability, and preference, scheme 6 (3D CAPTCHA with rotation and character color variation) is the best from our ten proposed schemes.

V. Discussion

In this section, the strength of our CAPTCHA method is analyzed in terms of the following:

A. Resistance to Pre-Processing

Pre-Processing [18] is the process of removing noise and complicated background from the CAPTCHA image. The background of the 3D CAPTCHA is randomly selected from many forms. Hence, it is difficult to expect which background is used. If the bots attempt to remove the background, then the character is also removed as show in Fig. 25.

Figure 25. Pre-Processing result.

B. Resistance to Vertical Segmentation

Vertical Segmentation [19] is the simplest segmentation attack carried out to isolate the characters in the CAPTCHA image to facilitate their recognition. The technique fails because the overlapping factor can be added to make characters overlapping to each other as shown in Fig. 26.

Figure 26. Overlapping Factor added to 3D CAPTCHA.

C. Resistance to Color-Filling Segmentation

This Segmentation [20] attack involves flood-filling the different connected components of the CAPTCHA image with different colors. The algorithm fails because the characters in our method are fragmented. The result is shown in Fig. 27. As can be seen in the figure, the algorithm identifies many connected components in the same character due to fragmentation of pixels.

Figure 27. Each character containing many connected components.

D. Resistance to Pixel-Count Attack

Although each character has been distorted into a different shape but the pixel counts of foreground color in each character is stall the same. This means that if the bots can segment the CAPTCHA image into individual character [21]. The character can also be identified by pixel count. Our method is not prone to this attack because the pixel count of the character is changeable with respect to the rotation angles in three axes. Table I shows a number of pixels of a character “A” for various angles.

TABLE I. Pixel count of character “A”.

Character	Angle (degree)			Pixels Count
	X	*Y*	*Z*	
	0	0	0	505
	15	0	0	574
	30	0	0	598
	45	0	0	595

E. Resistance to Character Recognition by using OCR

OCR is a software tool for converting printed materials into text or word processing files that can be easily edited and stored. This tool is used to recognize a character after segmentation has been done. ABBYY FineReader 10 [22] is chosen to test this resistance. Scheme 1 is selected to be test because all other schemes are based on scheme 1 and share the same properties that inhibit OCR recognition. The result is shown in Table II.

TABLE II. OCR result.

Recognition Type	Recognition Result (percent)
Recognition per Character	5 %
Recognition per Word	0 %

F. Resistance to Dictionary Attack

Dictionary Attack is a technique that uses words from dictionary to break the CAPTCHA. Use of dictionary words makes analysis much easier. When OCR can extract the majority of the characters, it is a simple matter to run the result through a spell checking library and choose the most likely suggestion. This is especially true as the words get longer. This technique fails because the characters are randomly generated from alphanumeric characters that including the letters and the numbers.

VI. CONCLUSION

In this paper, we propose a new generation of the CAPTCHA method that uses 3D character instead of 2D character. This method based on assumption that human can recognize 3D character image better than Optical Character Recognition (OCR) software bots.

The advantage of using 3D characters in the CAPTCHA is it can recognizable by human users and difficult to read by bots. Our new 3D CAPTCHA method use a same input method similar with the other many well known web sites and services where users type some keywords or characters into an input box. Thus it is easy to learn and use by any user. The algorithm of this method makes it hard to read by OCR programs which mean that it is more secure.

We can increase the rate of its difficulty in order to improve its resistance against the attacks by applying other effects such as increase the length of the CAPTCHA, increase the rotation angle, or include more special characters.

REFERENCES

[1] A. Turing, “Computing machinery and intelligence,” Mind, vol. 59, no. 236, pp. 433-460, October 1950.

[2] L. von Ahn, M. Blum, and J. Langford, “Telling humans and computers apart automatically,” Communications of the ACM, vol. 47, no. 2, pp. 57-60, February 2004.

[3] M. Shirali-Shahreza and S. Shirali-Shahreza, “Collage CAPTCHA,” Proc. IEEE Int’l Symp. Signal Processing and Applications (ISSPA 2007), February 2007.

[4] T.Y. Chan, “Using a text-to-speech synthesizer to generate a reverse turing test,” Proc. IEEE Int’l Conf. Tools with Artificial Intelligence (ICTAI 2003), pp. 226-232, November 2003.

[5] Spamfizzle, http://spamfizzle.com/CAPTCHA.aspx.

[6] CNET news, http://news.cnet.com/8301-17938_105-10204300-1.html.

[7] Steam Forums, http://forums.steampowered.com/forums/search.php.

[8] Google Accounts, https://www.google.com/accounts/NewAccount.

[9] Hotmail, https://signup.live.com/signup.aspx.

[10] Friendster, http://www.friendster.com/join.php.

[11] eBay, https://scgi.ebay.com/ws/eBayISAPI.dll?RegisterEnterInfo.

[12] Yahoo! Registration, https://edit.yahoo.com/registration.

[13] Rediffmail Free Unlimited Storage, http://register.rediff.com/register/register.php.

[14] pc1news, http://www.pc1news.com.

[15] JEANS SMS, http://www.jeans.com.ua/ua/sms.

[16] PHP-CAPTCHA-Class, http://www.nogajski.de/priv/php/captcha.

[17] K. Larson, M. van Dantzich, M. Czerwinski, and G. Robertson, “Text in 3D: some legibility results,” Proc. Human Factors in Computing Systems Conf., pp. 145-146, 2000.

[18] G. Mori and J. Malik, “Recognizing objects in adversarial clutter: breaking a visual CAPTCHA,” Proc. IEEE Computer Society Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 134-141, June 2003.

[19] G. Moy, N. Jones, C. Harkless, and R. Potter, “Distortion estimation techniques in solving visual CAPTCHAs,” Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 23-28, June-July 2004.

[20] J. Yan and A.S. El Ahmad, “A low-cost attack on a microsoft CAPTCHA”, Proc. Computer and Communications Security Conf., ACM, pp. 543-554.

[21] K. Chellapilla and P. Simard, “Using machine learning to break visual human interaction proofs (HIPs),” in Advances in Neural Information Processing Systems, vol. XVII, L. K. Saul, Y. Weiss, and L. Bottou, Eds. MIT Press, 2004, pp. 265-272.

[22] ABBYY FineReader 10, http://finereader.abbyy.com