International Journal of Image Processing and Vision Science

Volume 1 | Issue 2 Article 14

October 2012

AN ALGORITHM TO ANALYZE STRENGTH OF CAPTCHA

AJINKYA KASHINATH PARBHANE
Dept. of Information Technology, MAEER's MIT, Pune, India, aparbhane@gmail.com

ANAJALI . A. CHANDAVALE
Dept. of Information Technology, MAEER's MIT, Pune, India, c.anjali38@gmail.com

A. M. SAPKAL
Dept. of Information Technology, MAEER's MIT, Pune, India, ams@extc.coep.org.in

Recommended Citation
PARBHANE, AJINKYA KASHINATH; CHANDAVALE, ANAJALI . A.; and SAPKAL, A. M. (2012) "AN ALGORITHM TO ANALYZE STRENGTH OF CAPTCHA," *International Journal of Image Processing and Vision Science*: Vol. 1 : Iss. 2 , Article 14.
DOI: 10.47893/IJIPVS.2012.1026
Available at: https://www.interscience.in/ijipvs/vol1/iss2/14

AN ALGORITHM TO ANALYZE STRENGTH OF CAPTCHA

AJINKYA KASHINATH PARBHANE, ANAJALI .A.CHANDAVALE & A.M. SAPKAL

Dept. of Information Technology, MAEER's MIT, Pune, India
E-mail : aparbhane@gmail.com, c.anjali38@gmail.com & ams@extc.coep.org.in

Abstract – CAPTCHA stands for Completely Automated Public Turing Tests to Tell Computers and Humans Apart. The CAPTCHAs have been widely used across the Internet to defend against undesirable and malicious bot programs. It was observed that an alarming number of CAPTCHAs could be broken by the technique of Image Processing and Artificial Neural Network. Many Researchers have tried to break a CAPTCHA so as to design robust CAPTCHA , but it is essential to generate a strong CAPTCHA that will resist bot attack. This paper has proposed algorithm to analyze the strength of CAPTCHAs using simple image processing techniques such as Preprocessing, Segmentation and Character recognition which in turn helps to improve the robustness and usability of CAPTCHA in Internet System. The experimental result shows the proposed algorithm gives 75 % accuracy to analyze the strength of CAPTCHA.

Keywords- *CAPTCHA Analyzing, Segmentation, and Character recognition.*

I. INTRODUCTION

A CAPTCHA (Completely Automated Public Turing Test to Tell Computers and Humans Apart) is a program that generates and grades tests that are human solvable, but beyond the capabilities of current computer programs [1].CAPTCHAs are widely used on the internet as a security measure to prevent bots from automatically spamming registration forms. CAPTCHA is image to ensure that each request comes from an individual human and is not an attempt by an automated program to access systems illegally. With the rapid development of internet, more and more websites utilize CAPTCHAs to protect against auto registration login, to prevent against spam ,comment in blogs, dictionary attacks and resist search engine bots.

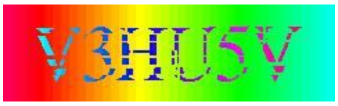


Figure 1. CAPTCHA Image with RGB color background and fuzzy character and number.

The robustness of a CAPTCHA is its strength in resisting adversarial attacks and improvement in its usability. It is proposed that a good CAPTCHA must be both robust and usable. Around the world various website have diverse CAPTCHA generation technique so there is no uniformity to analyze the CAPTCHA whether it is strong or weak to defend the bots attack. Solving CAPTCHA is a Hard AI image processing problem in the general case [2] so it has attracted considerable attention in the research community. To solve the problem the paper has proposed algorithm that is applicable to any CAPTCHA used in web world and investigates its strength and suggests recommendation to improve quality of CAPTCHA design. The rest of paper is organized as follows: Previous work about breaking of CAPTCHA in section 2. Section 3 gives implementation details of algorithm used to analyze CAPTCHA. The Testing results are given in Section 4 and finally section 5 concludes the paper.

II. PREVIOUS WORK

In 2003, Mori and Malik [3] proposed a shape matching algorithm to break EZ-Gimpy and Gimpy CAPTCHAs. They achieved a success rate of 92% in case of EZ-Gimpy and 33% in case of Gimpy. In 2004, Moy et.al. [4] use distortion estimation technique to break E-Z Gimpy CAPTCHAs and achieved a great success rate. Till date there are various researches on breaking the CAPTCHA. Among these TessarCap software is free software develop by MacAfee to break the CAPTCHA, recently in 2012, Gursev Singh Kalra used TesserCap software to break visual text based CAPTCHA to evaluate CAPTCHA strength but the Tessar Cap software has more manual interference and time consuming process to evaluate the images [5].As we compare with proposed algorithm, observes that proposed algorithm is more durable to used and efficient for analysis purpose.

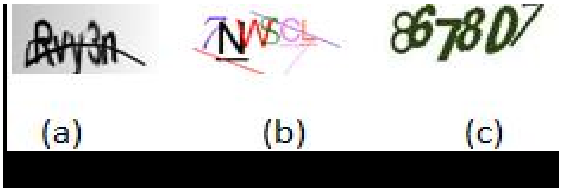


Figure- 2(a)- Nokia CAPTCHA, (b)- Maharashtra state Labour office website, (c) - EBay CAPTCHA

There are several CAPTCHA images having more diversity in noise level, transformation of Character, twisted and wavy line .As there is no unique method to break these CAPTCHA so this paper will proved a solution to over come problem.

III. IMPLEMENTATION

The Proposed algorithm has different phases such as preprocessing, segmentation and character recognition which are explained in details as mentioned below

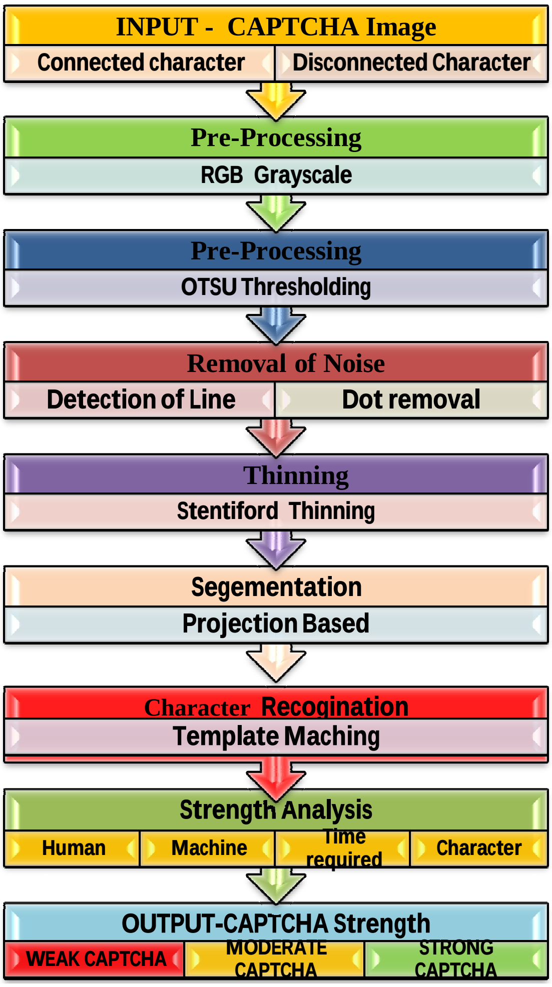
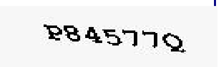
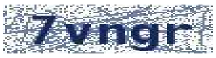
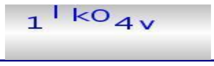
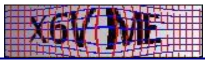


Figure 3: System architecture.

The samples are taken from real time dynamic website and strength of sample determine at end of implementation.

Table I: Original CAPTCHA Samples

Sample	CAPTCHA Instance	Website referred
Type 1		www.irctc.com
Type 2		www.ibps.com
Type 3		www.idea.com
Type 4		www.incometax.com.in

3.1 Pre-Processing

The Pre-processing is summation of all the operation like conversion of color to grayscale, then converted to binary format .As the image has noise in background hence it is necessary to eliminate in preprocessing phases and reduces noise level, thereby simplifying processing for the rest of stages. The main goal of noise removal is to remove unwanted bit pattern, which doesn’t reflect change in final output. Further preprocessed clean image is to get ahead for the segmentation phase.

3.1.1 Graying

CAPTCHA image contains many combinations of colors and to work on each of this color is very difficult so it necessary to convert into gray scale by provide a way to work on 256 intensity values[12].

Algorithm:

- Accept CAPTCHA Images in JPEG as input.
- Obtain pixel values by using grab Pixel () function.
- Obtain gray scale value for each RGB value by using the formula
- $G = (.56 * g + .33 * r + .11 * b)$

  Where r, g, b are the red, green and blue color components of the pixel in the image.
- For each pixel in the input image, replace the RGB value by its corresponding gray scale value.
- Stop.

3.1.2. Thresholding

The Thresholding is to convert 8-bit grayscale image to black-and-white image[9]. There are two objectives to threshold. One is to decrease subsequent data processing, and the other is to extract the objects from the background. Generally it is observed that CAPTCHA is composed of dark objects on a light back ground [6]. One way to extract the objects from the background is to select a threshold T. Any point(x, y) for which f(x, y) < T is called an object point; otherwise, the point is called a background point. T selection [8] is described by

$$T=T\,[x, y, p(x, y), f(x, y)] \qquad (1)$$

Where f(x, y) is the gray level of point(x, y) and p(x, y) denotes some local property of this point, the average gray of a neighborhood centered on (x, y). When T depends only on f(x, y), the threshold is called global. If T depends on f(x, y) and p(x, y), the threshold is called local. If T depends on the spatial coordinates x and y, the threshold is called dynamic [6]. The pixel's value g(x, y) after thresholoding is define s

$$G(x, y) =\{ 255 \ f(x ,y) > = T \text{ background}$$

$$f(x, y) \ T < \text{object} \qquad (2)$$

The key parameter in the thresholding is to find a global or multiple local optimized threshold value T. The proposed algorithms select value of T by using OTSU threshold. OTSU threshold based on class variance. It determines the maximize threshold between-class variance. It computes threshold value as

$$(T) =W_a (\mu a-\mu)^2 \ +W_{b} (\mu b-\mu)^2 \qquad (3)$$

Where (*T*) is a threshold maximizing the between-class variance; *Wa* and µa are the probabilities of class object occurrence and object mean level ; *Wb* and µb are the probabilities of class background occurrence and background mean level; µ is the total mean level of the original picture. Finally threshold (*T*) value is obtained.

Binary images are often produced by thresholding a grayscale or color image, in order to separate an object in the image from the background. The color of the object (usually white) is referred to as the foreground color. The rest (usually black) is referred to as the background color.

3.1.2 Removal of Noise

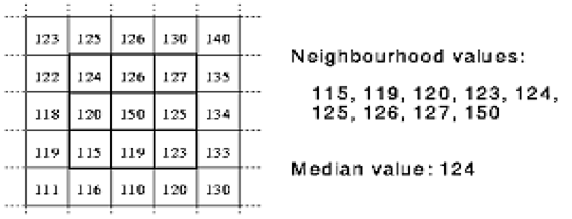
The Noise level in the CAPTCHA comprised of line, dots and wavy line in between character to increase its strength. In order to obtained clean image removal of line and dots are necessary.

A)Line removal

The CAPTCHA images sometimes contain horizontal lines and vertical lines. To remove these lines the number of continuous black pixels in row or columns is counted. If the count is more than 80% of total width or height of the image, then detected as a line and thus removed it by making it white.

B) Median Filter

The median filter is applied to reduce noise level in respect of smoothing of character and removal of unnecessary dots present in the image. As the lines are removed, characters become discontinuous. To overcome this discontinuity the paper use the 8-connected technique to detect a white pixel which is surrounded by at least one black pixel in its 8-connected region, if this condition is true then we convert the white pixel to black. To remove unnecessary dots present in the image, the image is scanned for calculating the median value. First sort all the pixel values from the surrounding neighborhood into numerical order and then replacing the pixel being considered with the middle pixel value. The median value must be the actually value of one of the pixels in the neighborhood, here the median value does not create any new pixel when the filter is applied and if the median value is less than a particular threshold then makes it white[6][7].



123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

Neighbourhood values:

115, 119, 120, 123, 124, 125, 126, 127, 150

Median value: 124

Figure 4: Median filter Calculation

3.2 Thinning

The Skeletonization was introduced to describe the global properties of objects and to reduce the original image into a more compact representation. A basic method for skeletonization is thinning. The proposed algorithm works on Stentiford Thinning process. The Stentiford Thinning uses the principle of Template mark-and-deleting which helps in creating a set of four 3 X 3 templates to scan the image.

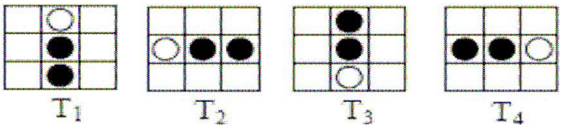


Figure 5. Templates to identify pixels to be eroded in theStentiford Method. The empty white boxes belong to Places where the color of the pixel does not need to be checked.

The Stentiford Algorithm can be stated as following [10]:

1. Find a pixel location (i, j) where the pixels in the image match those in template T1. With this template all pixels along the top of the image are removed moving from left to right and from top to bottom.
2. If the central pixel is not an endpoint, and has connectivity number = 1, then mark this pixel for deletion.
3. Repeat steps 1 and 2 for all pixel locations matching T1.
4. Repeat steps 1-3 for the rest of the templates: T2, T3, and T4.

T2 will match pixels on the left side of the object, moving from bottom to top and from left to right. T3 will select pixels along the bottom of the image and move from right to left and from bottom to top. T4 locates pixels on the right side of the object, moving from top to bottom and right to left.

5. Set white pixels for deletion.

Endpoint pixel - A pixel is considered an endpoint if it is connected to just one other pixel. That is, if a black pixel has only one black neighbor out of the eight possible neighbors.

Connectivity number - It is a measure how many objects are connected with a particular pixel.

$$C_n = \sum_{k \in S} N_k - (N_k \cdot N_{k+1} \cdot N_{k+2}) \qquad (3)$$

Where: **Nk** is the color of the eight neighbors of the pixel analyzed. N0 is the center pixel. N1 is the color value of the pixel to the right of the central pixel and the rest are numbered in counterclockwise order around the center. Fig. 6 illustrates the connectivity number

S = {1, 3, 5, 7}

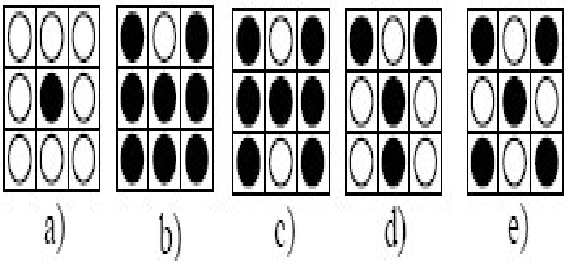


Figure 6 a) Represents connectivity number = 0. b) Represents connectivity number = 1, the central pixel might be deleted without affecting the connectivity between left and right. c) Represents connectivity number = 2, the deletion of the central pixel might disconnect both sides. d) Represents connectivity number = 3, and e) Represents connectivity number = 4

3.3 Segmentation

Here the image is segmented to extract letter from the CAPTCHA word. For each segmented character, it is thinned and scaled to a uniform size depend on the image size. The projection segmentation technique is implemented in the present the paper[11]. The traditional projection based algorithm works on fixed threshold, which makes it static. It focuses upon projecting the image data onto the X-axis. it is implemented by summing the number of non-white pixels in each column of the image parallel to the Y-axis as shown in Fig7.

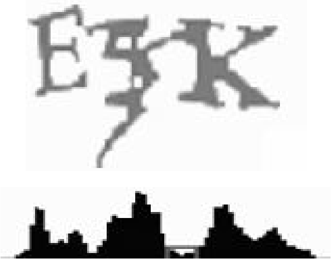
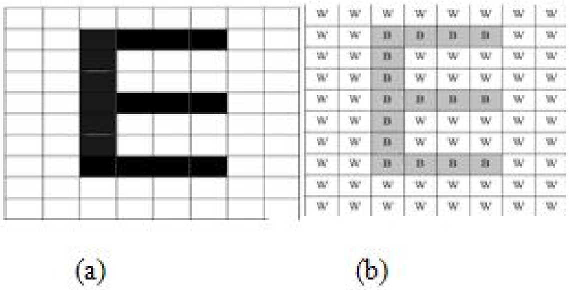


Figure 7 Projection Segmentation

The count black pixels in each column are compared with the threshold value and accordingly character is separated from each other in the uniform size. The segmented characters are pass further for character recognition.

3.4 Character Recognition with Template Matching

The character recognition algorithm has two essential components feature extractor and the classifier. The feature extractor derives the features that the character possesses. The derived features are then used as input to the character classifier. Template matching is one of the most common classification methods. Classification is performed by comparing an input character with a set of templates from each character class. In each comparison results is based on in a similarity measure between the input characters with a set of templates[11]. The amount of similarity increases when a pixel in the observed character is identical to the same pixel in the template image. If the pixels differ the measure of similarity it may decrease chance of matching. After all templates have been compared with the observed character image, the character's identity is assigned the identity of the most similar template. Template matching is a trainable process as template characters can be changed. The extraction of the character into a corresponding matrix is shown .here w= Numerical value corresponding to white and b= Numerical value corresponding to black. From this matrix algorithm find the character which matches the template the most and finally detects the character .



w	w	w	w	w	w	w	w
w	w	b	b	b	b	w	w
w	w	b	w	w	w	w	w
w	w	b	w	w	w	w	w
w	w	b	b	b	b	w	w
w	w	b	w	w	w	w	w
w	w	b	w	w	w	w	w
w	w	b	b	b	b	w	w
w	w	w	w	w	w	w	w
w	w	w	w	w	w	w	w

Figure 8 (a) Segementated character 'E" (b) Template Matching Matrix

3.5 Strength Analysis

Recognition of character is done by OCR and passed further for strength analysis. The basic principle of CAPTCHA is to distinguish between human and machine, hence the algorithm provide same CAPTCHA image is presented to human being and same time given to CAPTCHA solver(machine) as input image. If values obtained from human being matches with CAP Solver (machine) then it results in weak CAPTCHA else it is strong CAPTCHA. The percentage of analysis for proposed algorithm depends on the correct recognition rate and time required for breaking CAPTCHA. CAPTCHA Solver is able to break the image in less time and detect 75% of character in images then it come under weak category

Table II: Percentage to Evaluate Strength of CAPTCHA.

IV. TESTING RESULTS AND PERFORMANCE ANALYSIS

The Proposed algorithm has analyzed nearly 200 CAPTCHA samples from various website. Out of standard database, 30% of the samples were weak CAPTCHA, 45 % were Moderate CAPTCHA and 25 % were Strong CAPTCHA with respect to strength analysis. The strength analysis rate can be improved by considering a set of patterns for each character while measuring CAPTCHA Strength. .The paper has calculated 75 % Accuracy to analysis the CAPTCHA. Lastly comparison of Tessarcap1.0 vs. Proposed Algorithm is done.

Table-III: Proposed Algorithm with operation on samples

From Table 1

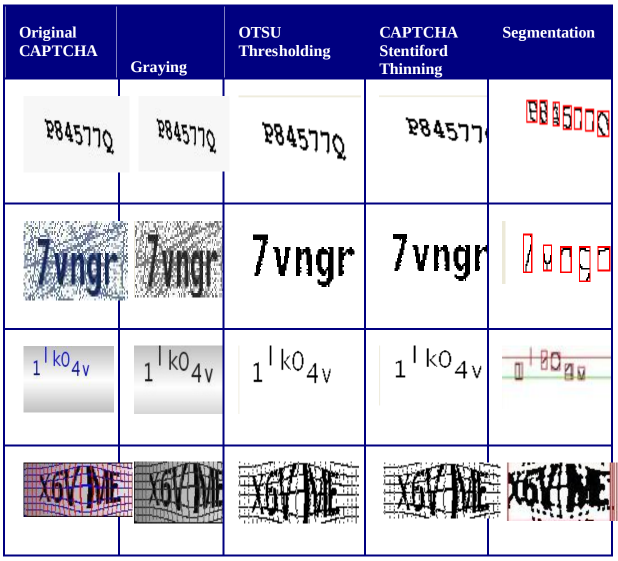
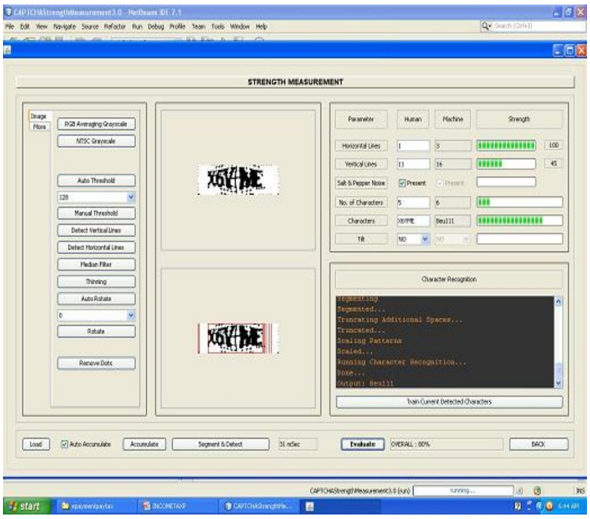

Original CAPTCHA
Graying
OTSU Thresholding
CAPTCHA Stentiford Thinning
Segmentation

Figure 8 –Type III CAPTCHA Sample are weak

Type –III CAPTCHA sample are easy broken by tool and its character are easy detected by the Machine in 16 msec.



Parameter	Human	Machine	Strength	
Horizontal Lines	1	3		100
Vertical Lines	11	16		45
Salt & Pepper Noise	☑ Present	☑ Present		
No. of Characters	5	6		
Characters	X6VME	8eu111		
Tilt	NO	NO		

```
Segmenting
Segmented...
Truncating Additional Spaces...
Truncated...
Scaling Patterns
Scaled...
Running Character Recognition...
Done...
Output: 8eu111
```

31 mSec

OVERALL : 80%

Figure 9: Type IV- CAPTCHA sample is 80 %Strong

Strength-Analysis	Human	CAPTCHA-Solver [Machine]
Detection of character in CAPTCHA	85-100 %	75 % - WEAK 50% - Moderate 25 % - Strong
Response Time to solve CAPTCHA	Second	Mill-Second

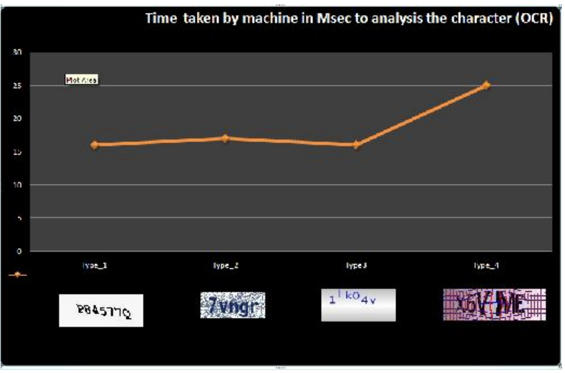


Figure 10 . Time taken by machine to analysis CAPTCHA

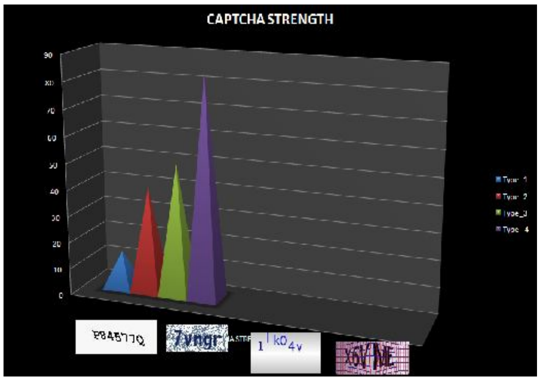


Figure11: CAPTCHA strength for four samples

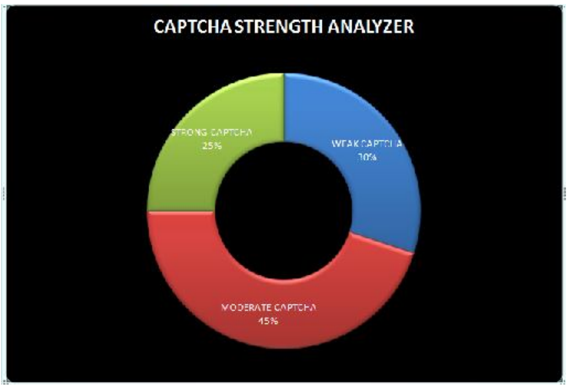


Figure12: CAPTCHA strength Analyzer showing that 30% weak CAPTCHA, 45% moderate, 25 % Strong CAPTCHA

Table IV: Tessar Cap 1.0 vs Proposed Algorithm

Parameter	Tessar Cap 1.0	Proposed System
Input Images	Website Link Must provide in URL.	No website link is required .Just to load user define CAPTCHA
Manual Interference	100 %	50 %
Response-Time for CAPTCHA Solver	Second	mill-Second
Graphics User Interface	Complex	Simple

V- CONCLUSION

The proposed algorithm has analyzed the strength of CAPTCHA that are deployed on Internet to resist attack of Bot. We conclude that Type-IV CAPTCHA samples are strong due to connected characters present in CAPTCHA image. The projection based algorithm used for segmentation gives accuracy of 98% for disconnected characters but it fails to segment connected characters. Our future work will concentrate on segmentation of connected characters. As per the CATPCHA designs we proposed valuable recommendation for CAPTCHA to improve its usability in online system. We conclude that proposed algorithm helps in building more secured CAPTCHA and maintain balance between authentication and security in online system.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their useful suggestions that helped in improving the quality of this paper. This work was supported in part by MAEER's MIT Pune.

REFERENCES

[1] L von Ahn, M Blum and J Langford. "Telling Humans and Computer Apart Automatically", CACM, V47, No2, 2004.

[2] L. von Ahn, M. Blum, N. Hopper, and J. Langford. CAPTCHA: Using Hard AI Problems for Security. In Advances in Cryptology EUROCRYPT 2003, page 646. 2003.

[3] G Mori and J Malik. "Recognizing objects in adversarial clutter: breaking a visual CAPTCHA", IEEE Conference on Computer Vision & Pattern Recognition (CVPR), 2003, IEEE Computer Society, vol. 1, pp.I-134-I-141, June 18-20, 2003.

[4] Gabriel Moy, Nathan Jones, Curt Harkless, and Randall Potter "Distortion Estimation Techniques in Solving Visual CAPTCHAs" proceedings of the Computer Vision and Pattern Recognition (CVPR'04) Conference ,IEEE Computer Society ,vol. 2 ,pp.23-28,2004.

[5] Gursev Singh Kalra Managing Consultant Foundstone Professional Services "Attacking Visual CAPTCHAs with TesserCap" white paper-2012

[6] Jisong Zhan, Xingfen Wang "Breaking Internet Banking CAPTCHA Based on Instance Learning", 2010 IEEE DOI 10.1109/ISCID.2010.18 pp 39-43.

[7] M. Gervaultz, W. Purgathofer. A Simple Method for Color Quantization: Octree Quantization. Graphics Gems. San Diego: Academic Press professional, pp. 287-293, 1990.

[8] R.C. Gonzalez and R. E. Woods. Digital Image Processing (Second Edition), pp.595-612, 2007

[9] N. Otsu. A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Systems, Man, and Cybernetics,Vol. 9, No. 1, pp. 62-66, 1979.

[10] Parker, J., R., "Practical Computer Vision using C", Wiley Computer Publishing, 1994.

[11] Sonka, M., Hlavac, V., Boyle, R., "Image Processing, Analysis, and Machine Vision", 2nd Edition, Pws. Pub. Co., 1998

[12] Prof.Mrs.Anjali.AChandavale, Dr.A.M.Sapkal "Algorithm for secured online authentication using CAPTCHA" Third International Conference on Emerging Trends and Engineering 2010 pp292-298