Hindawi
Security and Communication Networks
Volume 2017, Article ID 6898617, 15 pages
https://doi.org/10.1155/2017/6898617

Review Article

A Survey on Breaking Technique of Text-Based CAPTCHA

Jun Chen,[1,2] Xiangyang Luo,[1] Yanqing Guo,[3] Yi Zhang,[1] and Daofu Gong[1]

[1] *State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450002, China*
[2] *Henan Institute of Science and Technology, Xinxiang 453003, China*
[3] *Dalian University of Technology, Dalian 116024, China*

Correspondence should be addressed to Xiangyang Luo; luoxy_ieu@sina.com

Received 25 September 2017; Accepted 27 November 2017; Published 24 December 2017

Academic Editor: Zhenxing Qian

The CAPTCHA has become an important issue in multimedia security. Aimed at a commonly used text-based CAPTCHA, this paper outlines some typical methods and summarizes the technological progress in text-based CAPTCHA breaking. First, the paper presents a comprehensive review of recent developments in the text-based CAPTCHA breaking field. Second, a framework of text-based CAPTCHA breaking technique is proposed. And the framework mainly consists of preprocessing, segmentation, combination, recognition, postprocessing, and other modules. Third, the research progress of the technique involved in each module is introduced, and some typical methods of segmentation and recognition are compared and analyzed. Lastly, the paper discusses some problems worth further research.

1. Introduction

As a multimedia security mechanism, CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart [1]) also called Human Interactive Proofs (HIP [2]), can protect multimedia privacy. Now, it has been successfully applied to Google, Yahoo, Microsoft, and other major websites. In order to verify security and reliability of CAPTCHA, the breaking technology came into being. It involves image processing, pattern recognition, image understanding, artificial intelligence, computer vision, and many other disciplines. The research on CAPTCHA breaking has great value in research and application. First of all, CAPTCHA breaking can verify the security of existing CAPTCHAs, and it can promote the development of CAPTCHA design technique. Secondly, the CAPTCHA is an integral part of artificial intelligence and an important prerequisite to actualize natural human-computer interaction. Finally, the research of breaking CAPTCHA not only constantly refreshes limits to Turing test, but also can be applied in other fields such as digital paper-based media, speech recognition, and image labeling.

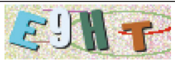
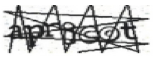
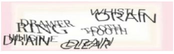
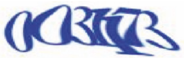
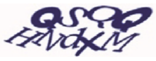
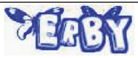
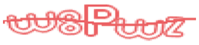
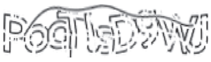
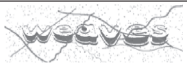
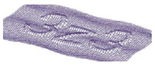
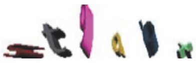
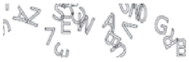
In recent decades, with the continuous development of CAPTCHA technology, relevant literature sources are abundant day by day. In 2013, [3] introduced CAPTCHAs of the time and attacks against them; the authors investigated the robustness and usability of CAPTCHAs at the time and discussed ideas to develop more robust and usable CAPTCHAs. Five years later, it is necessary to reorganize the emerging literature sources. Based on the research of text-based CAPTCHA breaking technique, this paper will review the relative research and prospect future trends.

The remainder of this paper is organized as follows: Section 2 briefly introduces the text-based CAPTCHA. Section 3 provides an overview of the text-based CAPTCHA breaking technique. Sections 4–8 describe main steps in the overall framework of the text-based CAPTCHA breaking technique. Section 9 points out some problems which can be further studied. Section 10 concludes up the full manuscript.

2. Overview on Text-Based CAPTCHA

In September 2000, the Carnegie Mellon University (CMU) research team designed the first commercial CAPTCHAs-Gimpy series text-based CAPTCHAs to resist malicious advertisements scattered by illegal scripting programs in the Yahoo chat room. At the same time, the research on CAPTCHA design and breaking also started. In 2002 and

Table 1: Typical types of text-based CAPTCHA and their features.

Type	Example	Source	Features
Solid CAPTCHA		Discuz!	Character independent, texture background, some interference
		Slashdot	A large number of interference lines and noise points
		Gimpy	Multiple strings, overlap, distortion
		Google	Unfixed length, distortion, adhesion
		Microsoft	Double-string, unfixed length, uneven thickness, tilting, adhesion
Hollow CAPTCHA		QQ	Hollow, shadows, interference shapes
		Sina	Hollow, adhesion, interference lines
		Yandex	Hollow, virtual contours, distortion, adhesion, interference lines
Three-dimensional CAPTCHA		Scihub	Hollow, shadows, interference lines, noise points
		Teabag	Grids, protrusion, distortion, background and character blending
		Parc	Colorful, shadow, rotation, zoom Special characters
Animation CAPTCHA		Program generating	Multiple characters jumping
		Hcaptcha	Multilayer character images blinking transformation

2005, the international seminars on HIP have been held, and a large number of related research results were published. In subsequent years, many research results were reported in international conferences including CVPR, NIPS, CCS, and NDSS. Many internationally renowned universities and research institutions have established research groups on CAPTCHA technology, such as CMU [1, 8–14], PARC [15–19], UCB [16, 17, 20, 21], Microsoft [2, 22–27], Google [28–30], Bell Laboratory [31, 32], Yan et al. [4, 33–42], Xidian University [41–47], and University of Science and Technology of China [48, 49]. In addition, many websites offer CAPTCHA services in public such as CAPTCHA [10], BotBlock [50], JCAPTCHA [51], and HCaptcha [52]. And some research groups focus on CAPTCHA recognition, such as PWNtcha [53], Captchacker [54], aiCaptcha [55], and Gery Mori [56].

The security of text-based CAPTCHA mainly depends on the visual interference effects [25], including rotation, twisting, adhesion, and overlap. The typical types of text-based CAPTCHA and their features are shown in Table 1.

To resist machine recognition, the text-based CAPTCHA's security is often protected by a series of technologies. From Table 1, we can sum up the following main features of the text-based CAPTCHA.

(1) A large enough character set. Only when a character set is large enough, the total number of CAPTCHA strings is large enough to resist violent breaking.

(2) The characters with distortion, adhesion, and overlap. Using characters with distortion, adhesion, and overlap, the breaking methods cannot easily segmented a CAPTCHA image into single characters.

(3) The characters are different in size, width, angle, location, and fonts. When comparing features of different characters, the various transformations may reduce recognition accuracy.

(4) The strings with unfixed length. In a CAPTCHA scheme, strings with unfixed length can increase breaking difficulty to a certain extent.

(5) Hollow characters and broken contours. Compared with the solid characters, hollow character's features are less, and broken contours can effectively resist the filling attack.

(6) The color and shape of complex backgrounds are similar to those of characters. If the images meet these conditions, the noise is difficult to remove. This may reduce recognition accuracy.

The above features effectively enhance text-based CAPTCHAs' security and bring great challenges to the CAPTCHA breaking research at the same time.

Table 2: Comparison of typical methods based on segmentation for breaking nonadherent CAPTCHA.

Example	Source	Success rate	Reference	Breaking method	Year
	Gimpy-r	78%	[57]	Segmentation: character gap Recognition: distortion evaluation	2004
	EZ-Gimpy	97.9%	[58]	Segmentation: connected region Recognition: distortion evaluation	2004
	Captcha-service	100%	[34]	Segmentation: vertical projection Recognition: statistical character pixels	2007
	Ego-share	92.2%	[5]	Segmentation: connected region Recognition: SVM	2009
	Ge-Captcha	100%	[59]	CW-SSIM	2010

Note. SVM: support vector machine, CW-SSIM: complex wavelet based structural similarity.

3. Research Progress of Breaking Text-Based CAPTCHA

For all kinds of text-based CAPTCHA schemes, the breaking methods are also various. According to whether there is segmentation or not, the existing breaking methods be contained in two categories.

3.1. Text-Based CAPTCHA Breaking Methods Based on Segmentation. The text-based CAPTCHA breaking based on segmentation has different processing methods for different objects and results. When there is no adherent character, individual characters are obtained using vertical projection and connected component with good effect. As shown in Table 2, the success rates of nonadherent character CAPTCHA range from 78% to 100%.

However, it had little success in adherent characters. Therefore, more complicated methods, such as different width, character features, and character contours, have been proposed one after another. With more and more antisegmentation technologies in CAPTCHA field, obtaining individual characters is becoming harder and harder. Then the researchers proposed the segmentation methods for obtaining character components by character structure, filters, and so forth. As can be seen from Table 3, the success rates of CAPTCHA breaking are generally low, with only a few higher than 80%.

3.2. Text-Based CAPTCHA Breaking Methods Based on Nonsegmentation. The text-based CAPTCHA breaking methods based on nonsegmentation can directly recognize preprocessed CAPTCHA images. The breaking method's success rate relies on recognition technique. In early stage, different pattern matching algorithms such as shape context [20] and similarity [57] are used for recognition. Later, with the improvement of the success rates of individual character recognition, researchers focus on the character segmentation technique. However, the text-based CAPTCHA design uses antisegmentation technique, which can prevent obtaining complete and individual characters. Nowadays with the advantage of deep learning, the breaking based on nonsegmentation will bounce back. The success rates of typical text-based CAPTCHA breaking methods based on nonsegmentation are as shown in Table 4.

3.3. The Framework of Text-Based CAPTCHA Breaking Technique. With the improvement of text-based CAPTCHA design, the breaking technique changes to meet it. The early text-based CAPTCHA contains nonadherent characters. The breaking technique is the traditional framework of "preprocessing + segmentation + recognition." In recent years, most of the text-based CAPTCHAs use CCT (Crowded Characters Together). Therefore, various breaking frameworks come into being, for example, "preprocessing + recognition," "preprocessing + recognition + postprocessing," "preprocessing + segmentation + combination + recognition," and "preprocessing + segmentation + combination + recognition + postprocessing."

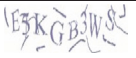
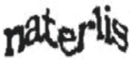
In this paper, the existing frameworks are integrated into an overall framework of text-based CAPTCHA breaking, as shown in Figure 1. The framework mainly consists of preprocessing, segmentation, combination, recognition, postprocessing, and other modules. The research progress of each module will be described in the following.

4. Preprocessing Methods of Breaking Text-Based CAPTCHA

The CAPTCHA preprocessing is the first step of CAPTCHA image processing before segmentation and recognition. Its main purpose is to highlight the information related to characters in a given image and to weaken or eliminate interfering information. The preprocessing of existing CAPTCHA breaking methods mainly includes image binarization, image thinning, denoising, and so on.

4.1. Image Binarization. Image binarization is to highlight interesting objects' contour and to remove noises in background. The key to binarization is to select an appropriate

Table 3: Comparison of typical methods based on segmentation for breaking adherent CAPTCHA.

Example	Source	Success rate	Reference	Breaking method	Year
	Google, Yahoo	4.89%–66.2%	[2]	Segmentation: width Recognition: CNN	2004
	Microsoft Google Yahoo	61% 8.7% 25.9%	[4]	Segmentation: color filling and projection Recognition: CNN	2008
	Hotmail	40%	[5]	Segmentation: change width Recognition: SVM Post-processing: DP search	2009
	MSN Yahoo	18% 45%	[6]	Segmentation: projection and central	2010
	Megaupload	78%	[36]	Segmentation: color filling Combination: nonredundancy Recognition: CNN	2010
	reCAPT- CHA Google	33% 46.75%	[38]	Segmentation: character structure feature Recognition: CNN	2011
	Yahoo	54.7%	[44]	Segmentation: projection and character feature Recognition: OCR	2012
	Yahoo	36%–89%	[41]	Segmentation: color filling Combination: redundancy Recognition: CNN Postprocessing: DFS	2013
	Microsoft	5.56% 57.05%	[60]	Different width/location segmenting and template matching	2015
	reCAPT- CHA	40.4%–94.3%	[61]	Segmentation: trichromatic code Recognition: SVM	2015
	Yahoo	57.3%–76.7%	[7]	Edge and fuzzy logic segmentation and recognition	2015
	Microsoft	5%–77.2%	[42]	Segmentation: Log-Gabor filter Combination: redundancy Recognition: KNN Postprocessing: DP search	2016
	MSN	27.1%–53.2%	[48]	Segmentation: different width Recognition: BPNN	2016

Note. CNN: convolutional neural network, DP: dynamic programming, OCR: optical character recognition, DFS: depth first search, KNN: *k*-nearest neighbor, BPNN: back-propagation neural network.

threshold. When the threshold is applied to the whole image, it is called the global threshold method; otherwise, it is called the local threshold method. If the threshold is not fixed during processing, it is called variable threshold method or dynamic threshold method. The common thresholding methods are Sauvola and Pietikainen's method [65], Otsu's method [66], and so on.

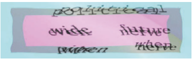
4.2. Image Thinning. Image thinning is to process the character's contour as skeleton. It must not change the character's adhesion. Its purpose is to highlight image contour and to simplify subsequent processing. The thinning algorithms contain two categories: noniterative algorithm and iterative algorithm. The common thinning algorithms include Hilditch algorithm [67] and Zhang and Suen algorithm [68].

4.3. Image Denoising. In order to resist breaking, there are noises and interference lines in CAPTCHA images. In addition, some noises are generated during grayscale and binarization. Therefore, we need to denoise CAPTCHA image. The typical methods are as shown in Table 5. We should choose the effective denoising method according to actual situation.

5. Segmentation Methods of Breaking Text-Based CAPTCHA

The segmentation aims to get individual characters or character components. There are the segmentation methods based on individual characters and the segmentation methods based on character components.

Table 4: Comparisons of typical methods based on nonsegmentation for breaking adherent CAPTCHA.

Example	Source	Success rate	Reference	Breaking method	Year
	EZ-Gimpy Gimpy	92% 33%	[20]	Shape context matching algorithm	2003
	EZ-Gimpy	99%	[57]	Correlation algorithm	2004
	Program generation	55%	[62]	RNN	2011
	Program generation	54.9%	[63]	2D LSTM-RNN	2013
	reCAPTCHA	99.8%	[30]	DCNN	2013
	reCAPTCHA	31.75%	[64]	HMM	2015

Note. RNN: recurrent neural network, 2D LSTM: 2-dimensional long short-term memory, DCNN: spatial displacement of the neutral network, HMM: Hidden Markov model.

5.1. Segmentation Methods Based on Individual Characters.

The segmentation methods based on individual characters segment a CAPTCHA image to individual characters. For individual characters, we can use segmentation methods based on character projection and connected components. For CCT characters, we can use segmentation methods based on character width, connected feature, and character contour.

5.1.1. Segmentation Methods Based on Character Projection.

The segmentation methods based on character projection determine the optimal segmentation position by analyzing the number of pixels projected under different conditions. This method applies to recognizing CAPTCHA characters without adhesion or slight adhesion. However, its effect is not obvious for the seriously adherent and distorted characters. The typical methods include vertical projection segmentation, horizontal projection segmentation, and guideline projection segmentation.

Using (1), [61] defines three-color bar code to segment reCAPTCHA images:

$$\text{Three-color Bar}(x)\begin{cases}\text{Blue}, & \text{for } H_{\Sigma}(x)=0,\\ \text{White}, & \text{for } H_{\Sigma}(x)=1,\\ \text{Black}, & \text{for } H_{\Sigma}(x)>1,\end{cases} \qquad (1)$$

where $H_{\Sigma}(x)$ represents the total of object pixels in the xth column. In three-color bar a column is colored in blue if there is not any pixel that belongs to character in the column ($H_{\Sigma}(x) = 0$). If there is only one pixel in column ($H_{\Sigma}(x) = 1$), the column is encoded by white. Finally, the black corresponds to the column with more than one object pixel ($H_{\Sigma}(x) > 1$), as shown in Figure 2(a). After denoising, the optimal segmentation line is determined in the middle of blue bar or white bar, as shown in Figure 2(b).

5.1.2. Segmentation Methods Based on Connected Components.

The segmentation methods based on connected components effectively segment individual characters using different connected components in an image. For slope and distortion characters, this method is effective. However, it is limited by adherent characters.

Reference [4] tried to segment Microsoft MSN CAPTCHA by combining connected components and vertical projection, as shown in Figure 3. First, different connected components are marked with different colors. And then the character blocks are generated according to different colors. Finally, strings are segmented to individual characters using the vertical projection feature, with a success rate of more than 90%.

5.1.3. Segmentation Methods Based on Character Width.

The segmentation methods based on character width are suitable for CAPTCHA images which are not easily segmented to individual characters. [60] used different widths (the average width of 0.75 times, 1 time, 1.5 times, and 2 times) to segment an image. Thus, each character corresponds to four recognition results, from which to find an optimal segment as the final recognition result. In addition, [5] did not take the average width as standard; they gave a set of character segments between the minimum width and the maximum width and then determined the optimal segmentation scheme using dynamic programming, as shown in Figure 4.

5.1.4. Segmentation Methods Based on Character Feature.

The segmentation method based on character features uses the features of CAPTCHA string, including inside features and outside features. Reference [38] classifies characters according to their own inside features, and each class contains the characters as shown in Table 6.

Reference [6] segments characters according to outside features among them. This paper proposes a new segmentation algorithm called middle-axis point separation

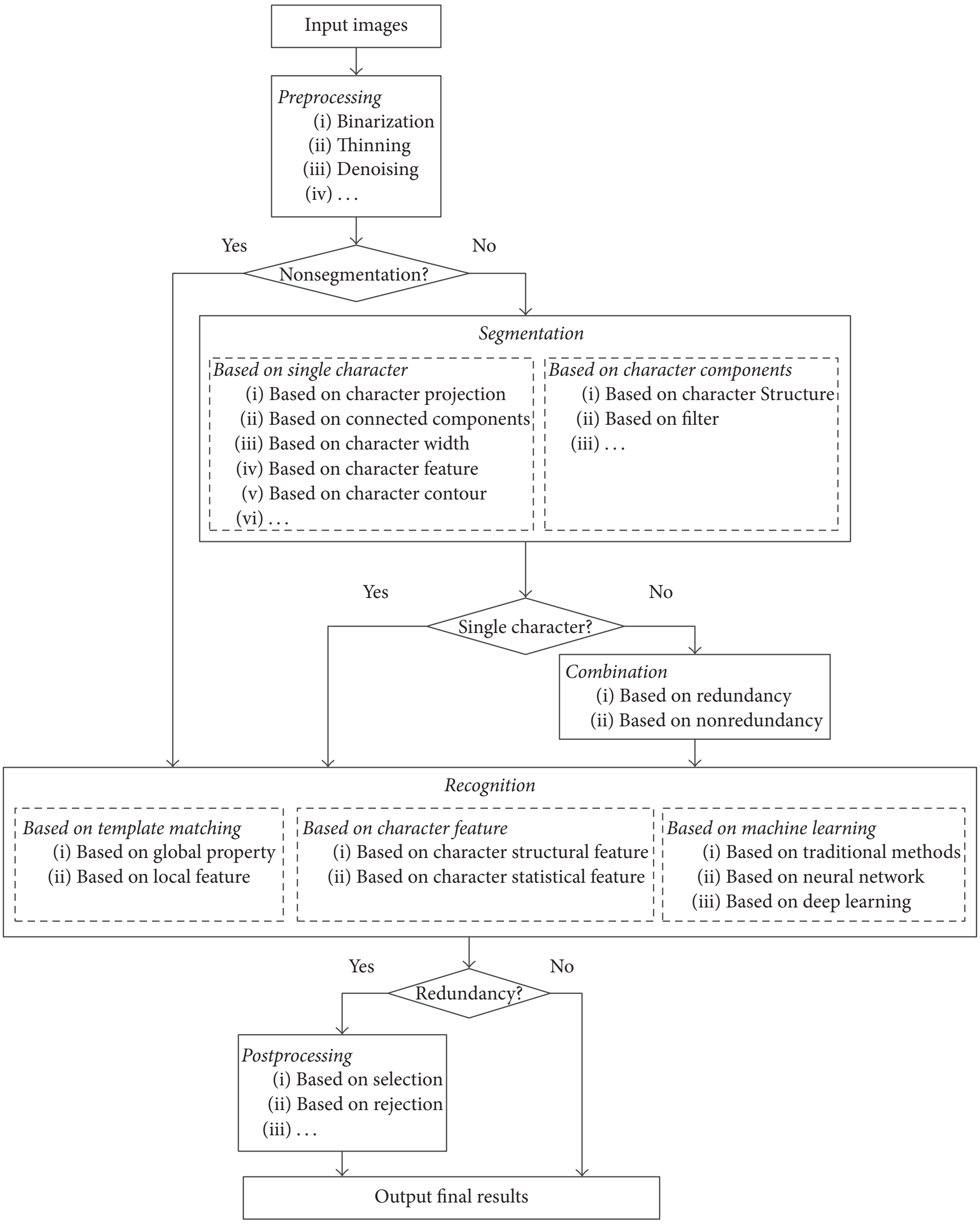
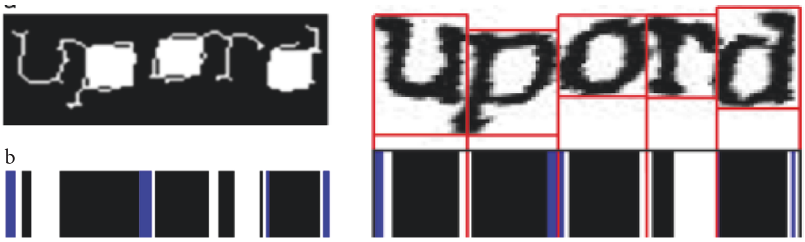

Input images
Preprocessing
(i) Binarization
(ii) Thinning
(iii) Denoising
(iv) . . .
Yes
No
Nonsegmentation?
Segmentation
Based on single character
(i) Based on character projection
(ii) Based on connected components
(iii) Based on character width
(iv) Based on character feature
(v) Based on character contour
(vi) . . .
Based on character components
(i) Based on character Structure
(ii) Based on filter
(iii) . . .
Yes
No
Single character?
Combination
(i) Based on redundancy
(ii) Based on nonredundancy
Recognition
Based on template matching
(i) Based on global property
(ii) Based on local feature
Based on character feature
(i) Based on character structural feature
(ii) Based on character statistical feature
Based on machine learning
(i) Based on traditional methods
(ii) Based on neural network
(iii) Based on deep learning
Yes
No
Redundancy?
Postprocessing
(i) Based on selection
(ii) Based on rejection
(iii) . . .
Output final results

Figure 1: The framework of text-based CAPTCHA breaking technique.



(a) Original three-color bar

(b) Denoised three-color bar

Figure 2: Three-color bar corresponding to CAPTCHA image.

Table 5: Comparisons of common denoising methods.

Denoising method	Typical algorithm	Implementation	Advantages	Disadvantages
Denoising method based on filter in the spatial domain	Average filter	The gray value of pixel is replaced by the mean of its neighboring pixels gray values.	The irrelevant details and gaps are removed.	The image is blurred.
	Median filter	The gray value of pixel is replaced by the median of its neighboring pixels gray values.	Remove effectively the salt and pepper noise, speckle noise.	Not applied to the image with many dots, lines, and spires.
	Wiener filter	The minimum mean square error criterion is used to adjust the filter effect.	Remove effectively the Gaussian noises.	Computation is complex.
Denoising method based on Gibbs and Hough transform	Gibbs	Markov random field theory.	Remove effectively noise points.	Not applied to irregular interference line.
	Hough transform	The straight line in the image is detected by using the point line duality of image space and Hough parameter space.	Remove effectively interference lines.	
Denoising method based on morphology	Open operation	First corrosion to expansion.	Smooth contours, cut off narrow lines, and eliminate fine.	The effect of denoising varies with operation mode and the size and shape of structural elements; the experiment needs to be repeated; the adaptability is poor.
	Close operation	First expansion to corrosion.	Smooth contour and fill holes, gaps, and fracture of contour line.	
Denoising method based on connected component	Connected component	The recursive method is used to find the connected domain to deal with pixel points, and then denoising based on gray features and morphological features of connected domain.	Remove effectively the noise interference, and the original details of the characters are generally not lost.	Need to analyze character's properties; hard to determine distinguish features.
Denoising method based on wavelet transform	Wavelet transform	Find the best mapping of original image in the wavelet transform domain to restore the original image.	Retain more image details.	Complex computation and it needs to adjust relative parameters.

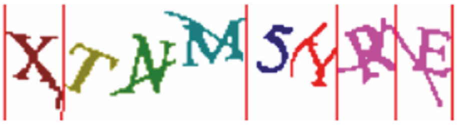


Figure 3: Segmented CAPTCHA image in [4].

Table 6: Character class table.

Class	Dot	Circle	Cross	S	V
Characters	i, j	a, b, d, e, g, o, p, q	t, f	s, z	v, w, y

for CAPTCHAs. The algorithm utilizes the central pixel in background between two disconnected object pixels as segmentation points (see Figure 5).

5.1.5. Segmentation Methods Based on Character Contour. The segmentation method based on character contours is to analyze geometric features of character contours, so as to determine the appropriate segmentation lines. Reference [7] tried to connect connection edge points between two merged characters and determined the optimal segmentation line by confidence, as shown in Figure 6.

5.2. Segmentation Methods Based on Character Components. The segmentation methods based on character components produce multiple character components, rather than individual characters. The segmentation methods are mainly base on character structure or filter.

5.2.1. Segmentation Methods Based on Character Structure. Using structural feature of characters with black components and white components, [36] segmented a seriously overlapped string to multiple components. First, locate black components, as shown in Figure 7(b). And then, locate white components, as shown in Figure 7(c). Finally, identify black components of each character and the shared white components.

In [41], a CAPTCHA image contains several hollow characters, whose contours naturally form several closed regions (see Figure 8(a)). According to this structural feature, a character is segmented to several character components by color filling (see Figure 8(b)).

5.2.2. Segmentation Methods Based on Filter. Reference [42] is the first to apply Gabor filters for breaking CAPTCHAs, which extracts character components along four directions by convolving a CAPTCHA image with each of four filters, respectively, as shown in Figure 9. The segmentation method is not limited by adhesion, distortion, and overlap and is suitable for many kinds of characters.

In summary, the contrast among segmentation methods is given. As can be seen in Table 7, each segmentation method applies to different types of characters. It is only the individualized segmentation method that can obtain good results.

6. Combination Methods of Breaking Text-Based CAPTCHA

An individual character after segmentation can be recognized directly. But character components need to be combined into an individual character to be recognized. According to the number of generated candidate characters, combination technologies can be divided into two categories: the combination technique based on redundancy and the combination technique based on nonredundancy.

6.1. Combination Methods Based on Redundancy. The number of candidate characters generated by combination technique based on redundancy is more than the number of real characters. In [42], each character fragment is labeled in order from top to bottom and left to right, and then the components are combined on the idea of jigsaw puzzle to generate candidate characters.

6.2. Combination Methods Based on Nonredundancy. The number of candidate characters generated by combination technique based on nonredundancy is equal to the number of actual characters. In [36], the character components are nonredundant. The overlap area strokes may be reused to compose a complete character. Figure 7(a) shows a Megaupload CAPTCHA image. Figure 10 gives the combined four characters. The final success rate of combination is 78.25%.

7. Recognition Methods of Breaking Text-Based CAPTCHA

Nowadays, the recognition methods used in text-based CAPTCHA system include three categories: template matching, character feature, and machine learning.

7.1. Recognition Methods Based on Template Matching. Template matching is to compare similarity of each pixel between characters and every template and to find the highest similarity. According to matching range, there are the matching recognition methods based on global property and the matching recognition methods based on local feature.

7.1.1. Matching Recognition Methods Based on Global Property. The matching recognition methods based on global property is traverse scanning. Within search area, the optimal match point to each pixel is found by regional correlation matching calculation. Because many templates matching each pixel will be pretty slow, [45] proposes the second template matching algorithm to improve efficiency. Only if a rough matching is successful, an exact matching needs to be made.

7.1.2. Matching Recognition Methods Based on Local Feature. The shape context is a simple local feature shape descriptor. Its basic idea is to convert the matching problem of image into the matching problem of feature point set. In 2003, Mori and Malik [20] used shape context to break the CAPTCHA of Gimpy and EZ-Gimpy. For good robustness to image

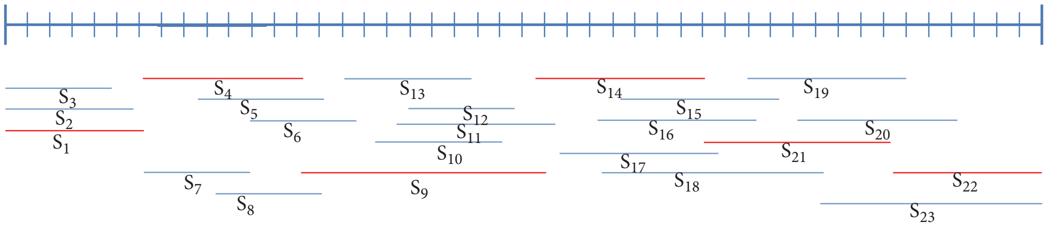


FIGURE 4: Character segments in [5].

TABLE 7: Comparisons of segmentation methods.

Segmentation methods	Main basis	Character features				Explanation
		Adhesion	Skew	Distortion	Overlap	
Segmentation based on individual character	Character projection	√	×	×	×	Suit strings with little adhesion rather than serious distorted, overlapped strings
	Connected components	×	√	√	×	Limited to overlapping rather than distortion
	Character width	√	×	×	×	Limited to severely distorted
	Character feature	√	×	×	×	Effective when character feature is notable and discriminant
	Character contour	√	√	√	×	Applied to the individual character with clear contour
Segmentation base on character components	Character structure	√	√	√	√	Only when character structure is easy to segment
	Filter	√	√	√	√	Wide application; the subsequent processing is complex

FIGURE 5: The middle-axis points in [6].

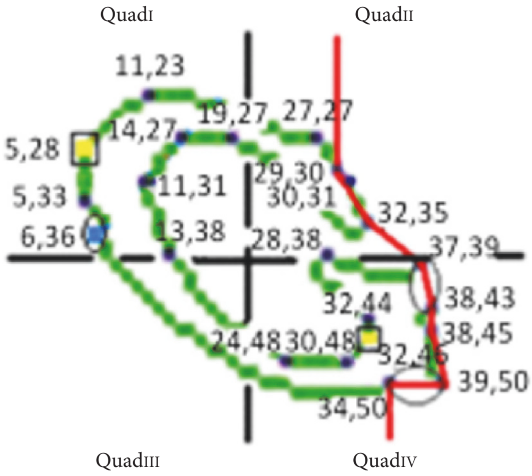


FIGURE 6: The optimal segmentation line in [7].

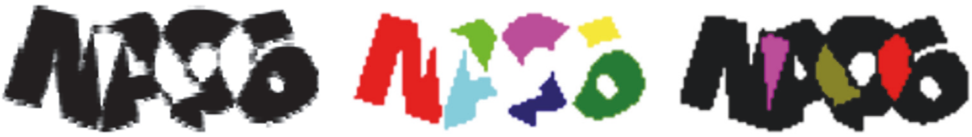
scaling and affine transformation, it is widely used in face recognition, CAPTCHA recognition, shape matching, and other fields.

7.2. Recognition Methods Based on Character Feature. Because the character of each CAPTCHA mechanism varies in design, we can define different methods according to the feature of characters, which is mainly based on character structural feature and character statistical feature.

7.2.1. Recognition Methods Based on Character Structural Feature. The structural feature can describe the details and structural information of characters, such as the number of loops, inflection point, convexo-concave degree, and cross points. For example, [46] uses the guidelines of characters (see Figure 11(a)) and closed loop detection (see Figure 11(b)) to break Yahoo CAPTCHA.

7.2.2. Recognition Methods Based on Character Statistical Feature. The recognition method based on character statistical feature uses commonly statistical features including pixel feature, projection feature, contour feature, and coarse mesh feature. This feature is robust to noise interference and is widely used in CAPTCHA recognition field. Reference [34] used the distinct pixel count for each of the letters A to Z (see Figure 12) to break captchaservice.org CAPTCHA with a near 100% success rate.

7.3. Recognition Methods Based on Machine Learning. The recognition methods based on machine learning is essentially using machine learning algorithms to correctly classify CAPTCHA characters. According to chronological order of

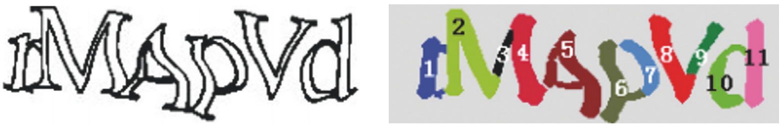


(a) Original image

(b) Nonshared character components

(c) Shared character components

Figure 7: An example of segmented CAPTCHA image in [36].



(a) Original image

(b) Segmented image

Figure 8: An example of segmented CAPTCHA image in [41].

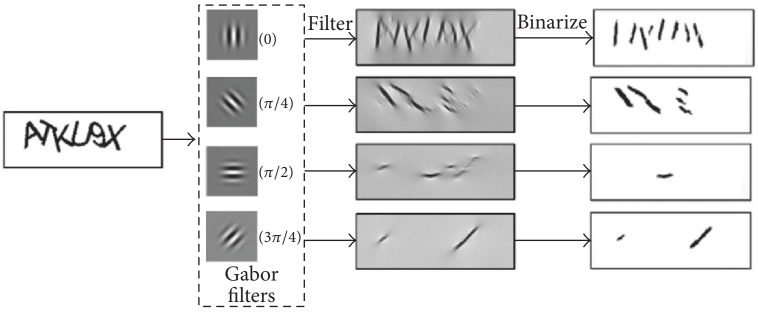


Figure 9: Segmentation CAPTCHA image by Gabor filters in [42].

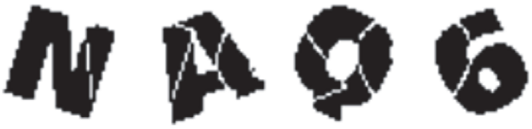


Figure 10: Individual characters after combination in [36].

mainstream, it can be basically divided into three categories: traditional methods, neural network, and deep learning.

7.3.1. Recognition Methods Based on Traditional Methods. In the field of text-based CAPTCHA recognition, the most widely used traditional classifiers include SVM and KNN.

The idea of SVM is to separate classes via a hyperplane. The key is kernel function, which is responsible for mapping original features into high-dimensional space in a nonlinear way, thereby improving the separability for data. Reference [5] compared four kernel functions: RBF (Radius-Based Function), POLY (polynomial), LINEAR, and SIGMOID. The experimental results showed that the performance of the first two kernel functions was optimal.

KNN is based on the category of the nearest K samples to determine the category of a sample. Reference [42] tested SVM, BPNN (back-propagation neural network), template matching, CNN, and KNN. Among these classifiers, KNN achieved higher success rates on most of the schemes, but CNN was faster most of the time.

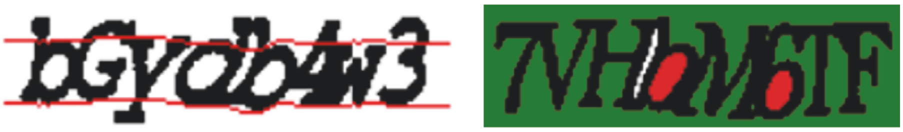
7.3.2. Recognition Methods Based on Neural Network. For the principle of parallel distributed operation in large number of neurons, the efficient learning algorithms, and the ability to imitate human cognitive systems, the neural network is very suitable to solve problems such as speech recognition and text recognition.

In [62], a BPNN used cross entropy for calculating the performance of a network with targets and outputs. Eventually, the system achieved an overall precision of 51.3%, 27.1%, and 53.2% for the CCT CAPTCHAs of Taobao, MSN, and eBay, respectively.

However, when applying neural network, we need to extract character features first. The quality of extracted features limits the final recognition rate to a certain extent.

7.3.3. Recognition Methods Based on Deep Learning. In recent years, deep learning has achieved remarkable achievements in recognition fields of text, image, audio, and so forth. The deep learning models commonly used in CAPTCHA recognition field are CNN, RNN, LSTM-RNN, and so forth.

CNN recognizes character images without feature extraction and has a certain degree of robustness in displacement, scale, and deformation. In the existing research results, a



(a) The guideline

(b) The closed loop detection

Figure 11: The example of character structural features in [46].

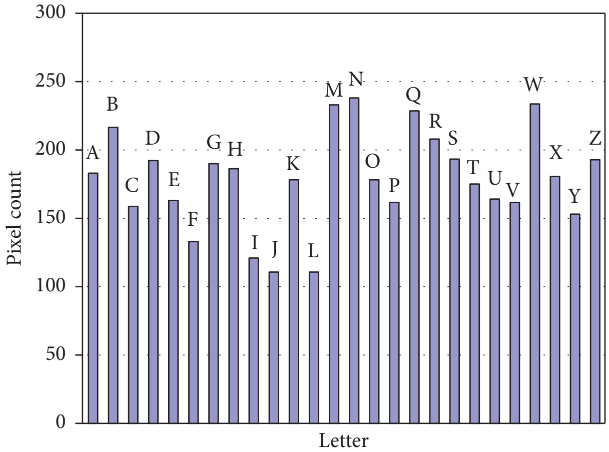


Figure 12: The pixel count for each of the letters A to Z in [34].

typical CNN is widely used [2, 4, 36, 38, 41] with a good recognition accuracy. Reference [30] trained large, distributed deep convolutional neural networks and achieved 99.8% accuracy in recognizing CAPTCHA images of reCAPTCHA.

However, due to lack of time dimension, CNN cannot combine context information in recognition. So RNN with feedback and time parameters was proposed to process time series data. Later, in order to solve vanishing gradient problem of RNN, LSTM was proposed in machine learning field. Reference [62] applied 2D LSTM-RNN in CCT CAPTCHAs recognition with a success rate of 55.2%. It innovatively obtained relative information not only in horizontal context, but also in vertical context.

In summary, a contrast among recognition methods is given, as shown in Table 8. According to the features of different networks, we should attempt to construct a new deep learning model by combining multiple networks. It will be a research trend in the field of text-based CAPTCHA recognition.

8. Postprocessing Methods of Breaking Text-Based CAPTCHA

In previous steps, some of character recognition results may be taken as final results directly, while others need to be further postprocessed. In postprocessing stage, the final result's reliability is ensured by simplification, selection, and optimization. According to different objects and methods, there are the postprocessing methods based on selection and the postprocessing methods based on rejection.

8.1. Postprocessing Methods Based on Selection. Usually, there are many redundant individual characters generated in previous steps. This requires selecting the most likely combined string as the final recognition result of CAPTCHA image. The selection strategies include the local optimization and the global optimization.

The local optimization selection only takes into account the recognition confidence optimality of an individual character. In [60], each character corresponds to several candidate characters with different widths. Therefore, the candidate character with the highest recognition confidence is selected as the final character.

The global optimization selection strives for the best results for all characters in an image. In [41], all candidate characters are found by the graph traversal, and then the string with the highest sum of characters recognition confidence values is taken as the final result, while in [5, 42], to avoid enumerating all candidate characters, a dynamic programming is used to determine the final result with the highest sum of characters' recognition confidence values directly. Compared with graph traversal, the dynamic programming is more effective and accurate.

8.2. Postprocessing Methods Based on Rejection. The purpose of postprocessing methods based on rejection is to determine whether the tested sample belongs to the types of training set by analyzing character recognition results. Therefore, the postprocessing methods based on rejection are a key to ensure high reliability of CAPTCHA recognition.

At present, the researchers have not been paid enough attention to the postprocessing methods based on rejection. To the best of our knowledge, there is only one paper [62] in CAPTCHA field. It considers multiple features, such as confidence, string length, character spaces, and the first and the last character of a string, to determine whether a candidate character should be rejected or not.

9. Some Problems Worth Further Research

As stated above, many achievements have been acquired. However, in view of the complexity of text-based CAPTCHA, there are still some issues worth exploring in depth in this field.

(1) Construction of Standard Test Database for Text-Based CAPTCHA. A rich and high quality text-based CAPTCHA image database is the necessary foundation for the research of text-based CAPTCHA breaking. At present, the researchers get CAPTCHA images mainly by web access and software

Table 8: Comparisons of recognition methods.

Recognition methods	Main basis	Typical methods	Advantages	Disadvantages
Recognition method based on template matching	Global property	Traversal search matching algorithm	The program is simple and suitable for standard character verification code.	The required template library is large; it depends on the choice of template matching.
	Local feature	Shape context matching algorithm	The image information is rich, and it is robust to image scaling and affine transformation.	Without rotation invariant.
Recognition method based on character feature	Character structure feature	Algorithm based on character structure feature	Sensitive to the details of characters; strong in distinguishing features.	The distortion is serious when there are noise interferences.
	Character statistical feature	Algorithm based on character statistical feature	Strong robustness against noise interference.	Targeted; application limited.
Recognition method based on machine learning	Template matching	SVM	Strong approximation ability and generalization ability; good adaptability and high accuracy for small sample space; suitable for two kinds of classification.	Not applied to infinite sample space.
		KNN	It is better to avoid the problem of imbalanced samples, which is suitable for overlapping samples of the same class.	Computation is complex; easy to misjudge in the domain with small sample size.
	Traditional method	BPNN	Flexible structure design, suitable for multiclass classification.	Slow convergence rate; depends on parameters.
	Deep learning	CNN	Accepts an input image directly; automatically extracts features; own robustness to displacement, scale, and deformation; high recognition accuracy.	Lack of time dimension; it could not identify using context information.
		RNN	Processes data in time series.	Time gradient may disappear.
		LSTM-RNN	Owns the time memory function; effective to prevent gradient disappear.	Unable to extract feature automatically.

generation. However, due to the diversity and timeliness of text-based CAPTCHA, it has not been possible to construct a common image database in the field of text-based CAPTCHA recognition. It is necessary to collect, classify, organize, and establish the text-based CAPTCHA images database. The database can provide the reliable training and testing data for research work and also provide the premise and basis of unified evaluation for various methods in this field.

(2) Multitype CAPTCHA Recognition. At present, only when training set and test set belong to the same type, the classifier can effectively recognize CAPTCHAs. In fact, there are a variety of character changes in a CAPTCHA. Therefore, it is an arduous and important task to design a reasonable classifier to recognize various types of CAPTCHAs.

(3) Segmentation-Free CAPTCHA Recognition. After more than ten years of development, the text-based CAPTCHA breaking has achieved a high success rate in individual character. However, the breaking success rate of the CAPTCHA string is generally low, and the results are less. With the wide application of CCT strings in text-based CAPTCHA, the problem of segmentation-free CAPTCHA recognition needs to be solved urgently. Now deep learning may provide new ideas and technical means to solve this problem.

(4) Application of Deep Learning Model. At present, in the field of CAPTCHA recognition, deep learning model can achieve better results than traditional methods. The most frequently used methods are based on CNN and its improved methods, while other deep learning models such as DBN (Deep Belief Networks), RNN, LSTM/BLSTM/MDLSTM, and DRL (Deep Reinforcement Learning) were not well used in text-based CAPTCHA recognition. Furthermore, the study of the interrelationships and fusion applications between the various deep learning models is not thorough. We hope that newer and better deep learning models are proposed to make a breakthrough in CAPTCHA recognition, which will certainly promote the development in this field.

(5) Rejection of Text-Based CAPTCHA. With the development of CAPTCHA breaking technique, the reliability of recognition results is also increasing. In this regard, on one hand, we should improve the correct rate of recognition; on the other hand, we should guarantee the correct rejection. In the field of CAPTCHA recognition, the concept of rejection has not been well known to the researchers. Therefore, this study has a potential development space.

(6) Misrecognition of Confusable Characters. When using the deep learning network to extract character features automatically, the characters with similar features are easily confused. It has practical significance to improve the precision of feature extraction and the training methods in the deep learning network.

10. Conclusions

Based on detailed investigation and in-depth analysis, this paper reviews the progress of text-based CAPTCHA breaking technique. First of all, this paper introduces various text-based CAPTCHAs and focuses on their features. Second, according to whether there is segmentation or not, we classify the existing breaking methods of text-based CAPTCHA and summarize their features. Meanwhile, we propose a framework of text-based CAPTCHA breaking technique and introduce the modules contained in the framework one by one. Next, we compare and analyze the basic principles, advantages, and disadvantages of the existing methods from five aspects: preprocessing, segmentation, combination, recognition, and postprocessing. Finally, some problems worth further research are discussed.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61379151, 61401512, 61572052, and U1636219), the National Key R&D Program of China (nos. 2016YFB0801303 and 2016QY01W0105), and the Key Technologies R&D Program of Henan Province (no. 162102210032).

References

[1] L. Von Ahn, M. Blum, and J. Langford, “Telling humans and computers apart automatically,” *Communications of the ACM*, vol. 47, no. 2, pp. 56–60, 2004.

[2] K. Chellapilla and P. Y. Simard, “Using Machine Learning to Break Visual Human Interaction Proofs (HIPs),” in *Proceedings of the Advances in Neural Information Processing Systems*, pp. 265–272, ofAdvances in Neural Information Processing Systems, 2004.

[3] N. Roshanbin and J. Miller, “A survey and analysis of current CAPTCHA approaches,” *Journal of Web Engineering*, vol. 12, no. 1-2, pp. 001–040, 2013.

[4] J. Yan and A. S. E. Ahmad, “A low-cost attack on a microsoft CAPTCHA,” in *Proceedings of the 15th ACM conference on Computer and Communications Security, CCS’08*, pp. 543–554, USA, October 2008.

[5] F. Jean-Baptiste and R. Paucher, “The Captchacker Project,” 2009, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.800.3065&rep=rep1&type=pdf.

[6] S.-Y. Huang, Y.-K. Lee, G. Bell, and Z.-H. Ou, “An efficient segmentation algorithm for CAPTCHAs with line cluttering and character warping,” *Multimedia Tools and Applications*, vol. 48, no. 2, pp. 267–289, 2010.

[7] R. A. Nachar, E. Inaty, P. J. Bonnin, and Y. Alayli, “Breaking down Captcha using edge corners and fuzzy logic segmentation/recognition technique,” *Security and Communication Networks*, vol. 8, no. 18, pp. 3995–4012, 2015.

[8] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford, “CAPTCHA: using hard AI problems for security,” in *Advances in cryptology—EUROCRYPT 2003*, vol. 2656 of *Lecture Notes in Computer Science*, pp. 294–311, Springer, Berlin, Germany, 2003.

[9] https://www.google.com/recaptcha.

[10] http://captcha.net/.

[11] http://www.captcha.net/captchas/bongo.

[12] A. Schlaikjer and A. Dual, "Use Speech CAPTCHA: Aiding Visually Impaired Web Users while Providing Transcriptions of Audio Streams," Tech. Rep. LTI-CMU-07-014, Carnegie Mellon University, Pittsburgh, Pa, USA, 2007.

[13] J. Tam, J. Simsa et al., "Improving Audio CAPTCHAs," in *Proceedings of the Symposium on Usable Privacy and Security*, 2008.

[14] J. Tam, S. Hyde, J. Simsa, and L. Von Ahn, "Breaking audio CAPTCHAs," in *Proceedings of the 22nd Annual Conference on Neural Information Processing Systems, NIPS 2008*, pp. 1625–1632, can, December 2008.

[15] H. S. Baird and K. Popat, "Human Interactive Proofs and Document Image Analysis," in *Proceedings of the International Workshop on Document Analysis Systems*, vol. 2423 of *Lecture Notes in Computer Science*, pp. 507–518, Springer, 2002.

[16] A. L. Coates, H. S. Baird, and R. J. Fateman, "Pessimal print: A reverse turing test," in *Proceedings of the 6th International Conference on Document Analysis and Recognition, ICDAR 2001*, pp. 1154–1158, usa, September 2001.

[17] M. Chew and H. S. Baird, "Baffletext: A human interactive proof," in *Proceedings of the Document Recognition and Retrieval X*, pp. 305–316, USA, January 2003.

[18] R. Chow, P. Golle, M. Jakobsson, L. Wang, and X. Wang, "Making CAPTCHAs clickable," in *Proceedings of the 9th Workshop on Mobile Computing Systems and Applications, HotMobile 2008*, pp. 91–94, USA, February 2008.

[19] P. Golle, "Machine learning attacks against the asirra CAPTCHA," in *Proceedings of the 15th ACM conference on Computer and Communications Security, CCS'08*, pp. 535–542, USA, October 2008.

[20] G. Mori and J. Malik, "Recognizing objects in adversarial clutter: breaking a visual CAPTCHA," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 134–144, June 2003.

[21] M. Chew and J. D. Tygar, "Image Recognition CAPTCHAs," in *Proceedings of the 7th International Information Security Conference*, vol. 3225 of *Lecture Notes in Computer Science*, pp. 268–279, Springer.

[22] K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski, "Designing human friendly human interaction proofs (HIPs)," in *Proceedings of the the SIGCHI conference*, p. 711, Portland, Oregon, USA, April 2005.

[23] P. Y. Simard, R. Szeliski, J. Benaloh, J. Couvreur, and I. Calinov, "Using character recognition and segmentation to tell computer from humans," in *Proceedings of the 7th International Conference on Document Analysis and Recognition, ICDAR 2003*, pp. 418–423, UK, August 2003.

[24] K. Chellapilla, K. Larson, P. Y. Simard, and M. Czerwinski, "Building segmentation based human-friendly human interaction proofs (HIPs)," in *Proceedings of the Second International Workshop on Human Interactive Proofs, HIP 2005*, pp. 1–26, usa, May 2005.

[25] K. Chellapilla, K. Larson, P. Simard, and M. Czerwinski, "Computers beat humans at single character recognition in reading based human interaction proofs (HIPs)," in *Proceedings of the 2nd Conference on Email and Anti-Spam*, usa, July 2005.

[26] J. Elson, J. R. Douceur, J. Howell, and J. Saul, "Asirra: A CAPTCHA that exploits interest-aligned manual image categorization," in *Proceedings of the 14th ACM Conference on Computer and Communications Security, CCS'07*, pp. 366–374, USA, November 2007.

[27] Y. Rui and Z. Liu, "ARTiFACIAL: Automated reverse turing test using FACIAL features," *Multimedia Systems*, vol. 9, no. 6, pp. 493–502, 2004.

[28] K. A. Kluever and R. Zanibbi, "Balancing usability and security in a video CAPTCHA," in *Proceedings of the 5th Symposium On Usable Privacy and Security, SOUPS 2009*, USA, July 2009.

[29] R. Gossweiler, M. Kamvar, and S. Baluja, "What's up CAPTCHA? A CAPTCHA based on image orientation," in *Proceedings of the 18th International World Wide Web Conference, WWW 2009*, pp. 841–850, Spain, April 2009.

[30] I. J. Goodfellow, Y. Bulatov, J. Ibarz et al., "Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks," 2014, https://www.researchgate.net/publication/259399973_Multi-digit_Number_Recognition_from_Street_View_Imagery_using_Deep_Convolutional_Neural_Networks.

[31] T.-Y. Chan, "Using a test-to-speech synthesizer to generate a reverse Turing test," in *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence*, pp. 226–232, Sacramento, Calif, USA, 2003.

[32] G. Kochanski, D. Lopresti, and C. Shih, "A reverse turing test using speech," in *Proceedings of the 7th International Conference on Spoken Language Processing, ICSLP 2002*, pp. 1357–1360, September 2002.

[33] http://www.lancaster.ac.uk/people/yanj2/.

[34] J. Yan and A. S. El Ahmad, "Breaking visual CAPTCHAs with naïve pattern recognition algorithms," in *Proceedings of the 23rd Annual Computer Security Applications Conference, ACSAC 2007*, pp. 279–291, December 2007.

[35] J. Yan and A. S. El Ahmad, "Usability of CAPTCHAs or usability issues in CAPTCHA design," in *Proceedings of the 4th Symposium on Usable Privacy and Security, SOUPS 2008*, pp. 44–55, July 2008.

[36] A. S. El Ahmad, J. Yan, and L. Marshall, "The robustness of a new CAPTCHA," in *Proceedings of the 3rd European Workshop on System Security, EUROSEC'10*, pp. 36–41, April 2010.

[37] B. B. Zhu, J. Yan, Q. Li et al., "Attacks and design of image recognition CAPTCHAs," in *Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS'10*, pp. 187–200, October 2010.

[38] A. S. E. Ahmad, J. Yan, and M. Tayara, "The Robustness of Google CAPTCHAs," Computing Science Technical Report CS-TR-1278, Newcastle University, 2011.

[39] A. S. El Ahmad, J. Yan, and W.-Y. Ng, "CAPTCHA design: Color, usability, and security," *IEEE Internet Computing*, vol. 16, no. 2, pp. 44–51, 2012.

[40] A. Algwil, D. Ciresan, B. Liu, and J. Yan, "A security analysis of automated Chinese turing tests," in *Proceedings of the 32nd Annual Computer Security Applications Conference, ACSAC 2016*, pp. 520–532, December 2016.

[41] H. Gao, W. Wang, J. Qi, X. Wang, X. Liu, and J. Yan, "The robustness of hollow CAPTCHAs," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, CCS 2013*, pp. 1075–1085, November 2013.

[42] H. Gao, J. Yan, F. Cao et al., "A Simple Generic Attack on Text Captchas," in *Proceedings of the Network and Distributed System Security Symposium*, pp. 1–14, San Diego, Calif, USA, 2016.

[43] http://web.xidian.edu.cn/hchgao/paper.html.

[44] H. Gao, W. Wang, and Y. Fan, "Divide and conquer: An efficient attack on Yahoo! CAPTCHA," in *Proceedings of the 11th IEEE International Conference on Trust, Security and Privacy*

in Computing and Communications, TrustCom-2012, pp. 9–16, June 2012.

[45] F. Dai, H. Gao, and D. Liu, "Breaking CAPTCHAs with second template matching and BP neural network algorithms," *International Journal of Information Processing and Management*, vol. 4, no. 3, pp. 126–133, 2013.

[46] H. Gao, W. Wang, Y. Fan, J. Qi, and X. Liu, "The robustness of "connecting characters together" CAPTCHAs," *Journal of Information Science and Engineering*, vol. 30, no. 2, pp. 347–369, 2014.

[47] H. Gao, X. Wang, F. Cao et al., "Robustness of text-based completely automated public turing test to tell computers and humans apart," *IET Information Security*, vol. 10, no. 1, pp. 45–52, 2016.

[48] R. Hussain, H. Gao, and R. A. Shaikh, "Segmentation of connected characters in text-based CAPTCHAs for intelligent character recognition," *Multimedia Tools and Applications*, pp. 1–15, 2016.

[49] R. Hussain, H. Gao, R. A. Shaikh, and S. P. Soomro, "Recognition based segmentation of connected characters in text based CAPTCHAs," in *Proceedings of the 8th IEEE International Conference on Communication Software and Networks, ICCSN 2016*, pp. 673–676, June 2016.

[50] https://captcha.com/.

[51] http://jcaptcha.sourceforge.net/.

[52] http://www.hinsite.com.

[53] http://caca.zoy.org/wiki/PWNtcha.

[54] https://code.google.com/p/captchacker.

[55] http://www.brains-n-brawn.com/default.aspx?vDir=aicaptcha.

[56] http://www.cs.sfu.ca/~mori/research/gimpy/.

[57] G. Moy, N. Jones, C. Harkless, and R. Potter, "Distortion estimation techniques in solving visual CAPTCHAs," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004*, pp. II23–II28, July 2004.

[58] A. Bansal, D. Garg, and A. Gupta, "Breaking a Visual CAPTCHA: A Novel Approach using HMM," 2008, https://pdfs.semanticscholar.org/3c2c/9af1e9a3b7095edaf8f205dfbadc30f-917fb.pdf.

[59] S. Li, S. A. H. Shah, M. A. U. Khan, S. A. Khayam, A.-R. Sadeghi, and R. Schmitz, "Breaking e-banking CAPTCHAs," in *Proceedings of the 26th Annual Computer Security Applications Conference, ACSAC 2010*, pp. 171–180, December 2010.

[60] C. Hong, B. Lopez-Pineda, K. Rajendran, and A. Recasens, "Breaking Microsoft's CAPTCHA," 2015, https://courses.csail.mit.edu/6.857/2016/files/hong-lopezpineda-rajendran-recansens.pdf.

[61] O. Starostenko, C. Cruz-Perez, F. Uceda-Ponga, and V. Alarcon-Aquino, "Breaking text-based CAPTCHAs with variable word and character orientation," *Pattern Recognition*, vol. 48, no. 4, pp. 1097–1108, 2015.

[62] L. Zhang, L. Zhang, S.-G. Huang, and Z.-X. Shi, "A highly reliable CAPTCHA recognition algorithm based on rejection," *Acta Automatica Sinica*, vol. 37, no. 7, pp. 891–900, 2011.

[63] R. Chen, J. Yang, R.-G. Hu, and S.-G. Huang, "A novel LSTM-RNN decoding algorithm in CAPTCHA recognition," in *Proceedings of the 3rd International Conference on Instrumentation and Measurement, Computer, Communication and Control, IMCCC 2013*, pp. 766–771, September 2013.

[64] S. Sano, T. Otsuka, K. Itoyama, and H. G. Okuno, "HMM-based attacks on Google's ReCAPTCHA with continuous visual and audio symbols," *Journal of Information Processing*, vol. 23, no. 6, pp. 814–826, 2015.

[65] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern Recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[66] 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.

[67] C. J. Hilditch, "Linear Skeletons from Square Cupboards," *Machine Intelligence*, pp. 403–420, 1969.

[68] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Communications of the ACM*, vol. 27, no. 3, pp. 236–239, 1984.