mathematics

MDPI

Article

Applying Visual Cryptography to Enhance Text Captchas

Xuehu Yan [1,*], Feng Liu [1], Wei Qi Yan [2] and Yuliang Lu [1]

[1] National University of Defense Technology, Hefei 230037, China; hemancute@163.com (F.L.); publicLuYL@126.com (Y.L.)

[2] Auckland University of Technology, Auckland 1142, New Zealand; dcsyanwq@gmail.com

* Correspondence: publictiger@126.com; Tel.: +86-055866927640

Received: 13 January 2020; Accepted: 26 February 2020; Published: 3 March 2020

Abstract: Nowadays, lots of applications and websites utilize text-based captchas to partially protect the authentication mechanism. However, in recent years, different ways have been exploited to automatically recognize text-based captchas especially deep learning-based ways, such as, convolutional neural network (CNN). Thus, we have to enhance the text captchas design. In this paper, using the features of the randomness for each encoding process in visual cryptography (VC) and the visual recognizability with naked human eyes, VC is applied to design and enhance text-based captcha. Experimental results using two typical deep learning-based attack models indicate the effectiveness of the designed method. By using our designed VC-enhanced text-based captcha (VCETC), the recognition rate is in some degree decreased.

Keywords: text captcha; visual cryptography; random grids; visual cryptography application; enhanced text captcha

1. Introduction

Nowadays, lots of applications and websites, including Baidu, Sina, Jingdong and many others, utilize text-based captchas to partially protect the authentication mechanism from certain types of attacks [1].

Text-based captchas belong to visual captchas. This type of captcha asks the user to identify some characters in an image deliberately rendered with some distortion and/or noise. Text-based captchas is widely used of since it is easily understood by most humans worldwide and since childhood we have been trained to recognize characters; it has a large brute force search space; its generation is easily automated without manual effort [2].

However, optical character recognition (OCR)-related techniques bring great challenges to text-based captchas. Text-based captcha enhanced methods and text-based captchas that automatically recognize ways—*a.k.a* breaking captchas [3–6]—are developed against each other.

However, by machine learning attacks, 2D text-based captchas are easily broken. More importantly, deep learning-based breaking ways [4–6] have posed great challenges for text-based captchas, such as convolutional neural networks (CNN) [6–8]. One typical breaking way [6] has a series of blocks including a convolution layer, pooling layer and a full connection layer connected to the output layer with an activation function.

Another typical deep learning breaking method is proposed in Reference [5] that includes four steps, that is, a captcha synthesizer, pre-processing, training and fine-tuning. A classic CNN model, namely LeNet-5 [9], is used in the method. It has an admirable performance on 33 captcha schemes including 11 schemes used by 32 of the top-50 popular websites, such as Wikipedia, eBay, Microsoft and Google. More importantly, the significance of the method proposed in Reference [5] is that it can achieve high performance only with small set of training captchas.

There are few studies in this area of enhanced text-based captchas, and the representative ones are as follows. In order to counter OCR and machine learning attacks, previous text-based captchas mainly use transformations and distortions. Transformation chiefly includes clockwise/counterclockwise rotation, translation and scaling, are easy for both humans and computers to solve. Then some level of distortions can be typically combined. The distortions can also be elastic deformations to the overall text (globalwarping) or deformations at the level of individual characters (localwarping).

The initial design idea of the popular "reCAPTCHA" was using distorted strings [10]. The MSN Passport captcha includes more characters highly warped to be distorted. Beheshti et al. [11] developed a model using the Human Visual System (HVS) to superimpose so as to integrate complex information presented in many frames. It utilized the concept of persistence of vision enabling humans to see the text in a continuous mode, which can use the brain to distinguish between automated computer bots and human users.

In addition to 2D text-based captcha, 3D text-based captcha was developed as well. Suzi et al. [12] introduced DotCHA based on human interaction, which allows the user to rotate a 3D text model to identify the text. It can show different letters according to the rotation angle. Unfortunately, it may be not easy to generate and identify.

In a word, traditional breaking captchas can be divided into two steps including segmentation and recognition. Thus, the above mentioned traditional captchas enhanced methods mainly focus on anti-segmentation and anti-recognition, respectively. Anti-segmentation chiefly includes overlapping characters, solid background, occluding lines, complex background and so on; anti-recognition includes rotation, distortion, waving, varied font sizes, styles and color, and so on. However, the recent development of deep learning-related ways threatens the security of text-based captchas. Thus, we have to enhance the text captchas design in a specific way, which is the motivation of this paper.

In this paper, due to the features of the randomness for each encoding process in visual cryptography (VC) and its visual recognizability with naked human eyes, VC is applied to design and enhance text-based captcha.

VC [13–15] for (k, n)-threshold encodes a binary secret image into several shadow images, *a.k.a.* shares or shadows, where each shadow image is random due to the randomness for the encoding process. When any k or more shadow images are collected, the secret image is restored by our human eyes without any computations (only Boolean OR operation), although some contrast loss appears; otherwise the number of the collected shadow images is less than k, the secret image cannot be restored even with high-performance computing devices and technologies [15,16]. In VC research field, random grids (RG)-based VC [17–19] is more feasible since RG-based VC has no pixel expansion and basic matrix design. VC can be applied to key management, password transmission [20], identity authentication [21,22], access control, digital watermarking [23–26], and distributive applications [27–29].

Since each shadow image is random, the randomness is preserved to some extent in the restored secret image. The randomness in VC and its visual recognizability with naked human eyes will be applied in this paper to design and enhance traditional text-based captcha, where the randomness is used to resist recognition and visual recognizability is served for humans. Random changes in the text-based captcha may invalidate a deep learning-based attack, which is the key idea of this paper.

In this paper, by utilizing the features of the randomness for each encoding process in visual cryptography (VC) and the visual recognizability with naked human eyes, we will apply VC to design and enhance traditional text-based captcha. Experimental results using two typical deep learning-based attack models indicate the effectiveness of the designed method. By using our designed VC-enhanced text-based captcha (VCETC) on the basis of traditional text-based captcha, the recognition rate is decreased in some degree.

The rest of the paper is presented as follows. Section 2 will introduce some preliminaries for the designed method. In Section 3, the designed method will be presented in detail. Section 4 will focus on experimental results.Finally, Section 5 will conclude this paper.

2. Preliminaries

In this section, we will introduce some preliminaries for the designed method. We summarize the main notations adopted in the paper in Table 1.

Table 1. The main notations used in this paper.

Notations	Descriptions
(k, n)	Threshold
0(resp.1)	A white(resp. black) pixel
S	The secret binary image
S'	The restored binary secret image
$AS\,0$ (resp. $AS\,1$)	The area of all the white (resp. black) pixels in S
$\overline{b}$	A bit-wise complementary operation on a binary pixel b
$\otimes$	Stacking (OR) operation
$\oplus$	Boolean XOR operation
$SC_1, SC_2, \cdots SC_n$	Shadow images generated by VSS schemes
t	Number of collecting shadow images in the recovery phase
α	Contrast of the restored secret image by stacking recovery
$P\left(x\right)$	The probability when any event x occurs
$A_1(A_2)$	A model trained with traditional captchas by the first (second) deep learning-based breaking method
$B_1(B_2)$	A model trained with captchas generated by our method by the first (second) deep learning-based breaking method

2.1. One Typical Text Captcha Generation Method

A typical enhanced text captcha generation method [1,6] generally includes occluding lines, overlapping, English letters and arabic numerals, complex background, varied font sizes and color, rotation, distortion and waving. The results of one traditional text captcha generation method are illustrated in Figure 1. Any previous-existing traditional text captcha generation method can be input in our design, and in this paper we will use this method as an example. For comparison, the CNN-based breaking methods will be separately trained on the captchas generated by the typical method [6] and by our method.

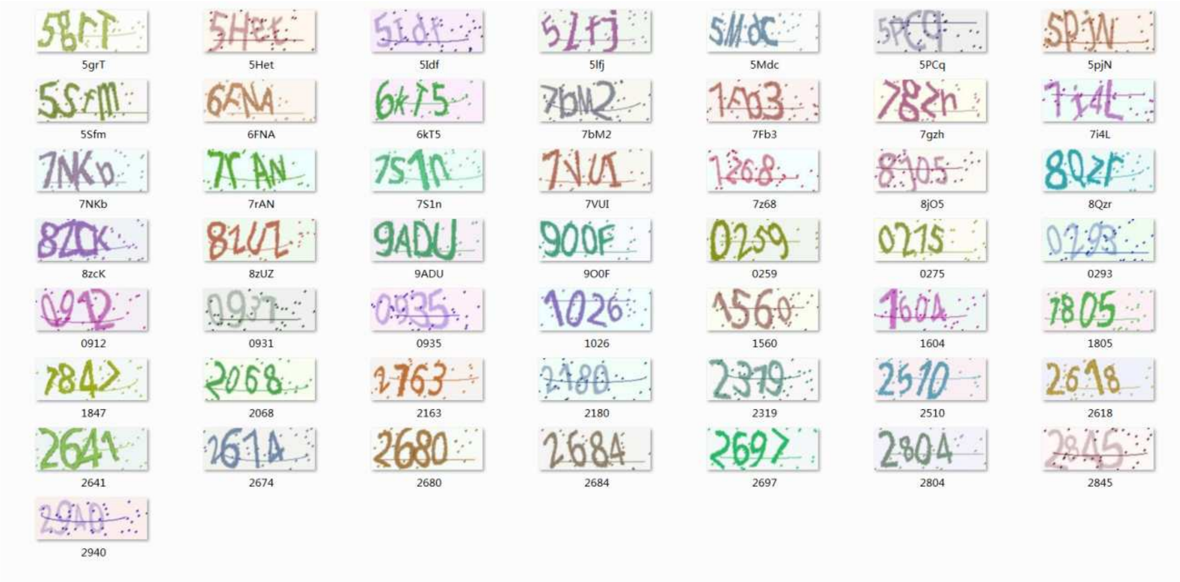


Figure 1. The captchas generated by one traditional text captcha generation method.

2.2. One Typical Deep Learning Breaking Method

Convolutional neural network (CNN) is a kind of deep learning network, which has achieved excellent results in many practical applications, such as image target recognition. One typical CNN architecture [6] is presented in Figure 2. A CNN architecture is divided into a series of blocks. The first is composed of two types of layers, that is, convolution layer and pooling layer, and the second one is the full connection layer connected to the output layer with activation function. The pooling layer is the pooling operation of data after the convolution layer is processed. CNNs are still layered networks, but the functions and forms of layers have changed. CNN is specifically designed for image recognition. Each image used in deep learning is divided into compact topological parts, each of which is processed with filters to search for specific patterns. The first CNN used in this paper includes two convolution layers and pooling layers, respectively.

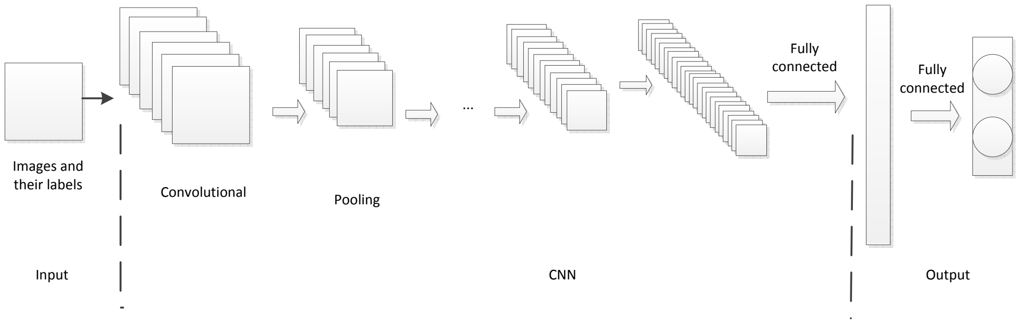


Figure 2. One typical convolutional neural network (CNN) architecture.

2.3. Another Typical Deep Learning Breaking Method

The framework of another typical deep learning breaking method [5] is realized by first learning a captcha synthesizer to generate synthetic captchas to learn a base solver. The base solver is then refined to obtain the final, fine-tuned solver with a few clean captchas. A classic CNN, namely LeNet-5 [9] is applied. The breaking method shows admirable performance on 33 captcha schemes including 11 schemes used by 32 of the top-50 popular websites, such as Wikipedia, eBay, Microsoft and Google. The significance of Reference [5] is that it can achieve high performance only with small set of training captchas. We will use the default parameters of Reference [5] in the experiment of this paper.

2.4. VC for (k, n)-threshold

Herein, symbols $\oplus$ and $\otimes$ denote the Boolean XOR and OR. $\bar{b}$ indicates bit-wise complementary operation of any binary bit b. A binary secret image S with size of $H \times W$ is split into n shadow images, denoted by $SC_1, SC_2, \cdots, SC_n$. The restored binary secret image S' is restored from any $t(2 \leq t \leq n, t \in Z^+)$ shadow images by superposing.

$AS\ 0$ (resp., $AS\ 1$) is the white (resp., black) area of S, that is, $AS\ 0 = \{(h,w)|S(h,w) = 0, 1 \leq h \leq H, 1 \leq w \leq W\}$ (resp., $AS\ 1 = \{(h,w)|S(h,w) = 1, 1 \leq h \leq H, 1 \leq w \leq W\}$).

For any pixel s of S, the probability of pixel color is transparent or white (0) is represented as $P(s = 0)$, and the probability of pixel color is opaque or black (1) is represented as $P(s = 1)$. In addition, $P(S = 0) = 1 - P(S = 1) = 1 - \frac{1}{HW} \sum\limits_{i=1}^{H} \sum\limits_{j=1}^{W} S(h,w), 1 \leq h \leq H, 1 \leq w \leq W.$

Definition 1 (Contrast)**.** *The image quality of the restored secret image S' is generally evaluated by contrast, denoted by α, as follows [19]:*

$$\alpha = \frac{P_0 - P_1}{1 + P_1} = \frac{P(S'[AS\ 0] = 0) - P(S'[AS\ 1] = 0)}{1 + P(S'[AS\ 1] = 0)}, \tag{1}$$

where P_1 indicates the wrongly restored probability for the black area of S and P_0 denotes the correctly restored probability for the white area of S.

Contrast is one of typical metrics [30] to evaluate the image quality of the restored secret image, which will be adopted in this paper. Contrast will decide how well human eyes can recognize the restored secret image. For clarity corresponding to different values of contrast [31], please refer to Figure 3. The value of contrast is in [0,1], which is expected to be as large as possible to achieve high image quality, where $\alpha = 0$ indicates S' has no relations with S and $\alpha = 1$ indicates $S' = S$, that is, S is losslessly restored.

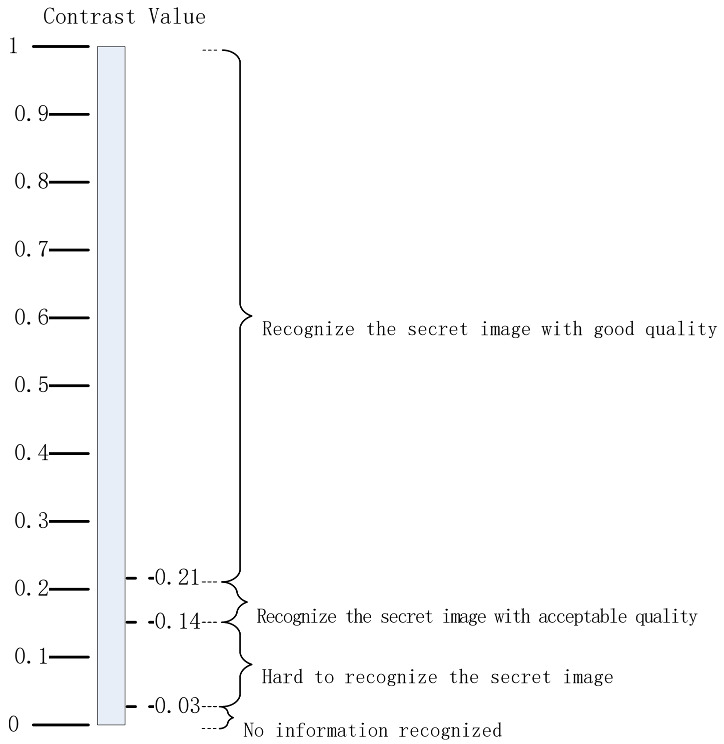

Contrast Value
1
0. 9
0. 8
0. 7
0. 6
0. 5
0. 4
0. 3
0. 2
0. 1
0
-0. 21
-0. 14
-0. 03
Recognize the secret image with good quality
Recognize the secret image with acceptable quality
Hard to recognize the secret image
No information recognized

Figure 3. Contrast value and its clarity in RG-based visual cryptography (VC).

The encoding and restoring phases of one typical $(2,2)$ RG-based VC [19] are reviewed in Algorithm 1.

```
Algorithm 1: One typical $(2,2)$ RG-based VC
    Input: A binary secret image $S$ with size of $H \times W$
    Output: 2 shadow images $SC_1$ and $SC_2$
    Step 1: For each secret position $(h,w) \in \{(h,w)|1 \leq h \leq H, 1 \leq w \leq W\}$, repeat Steps 2-5
    Step 2: Generate $b_1, b_2 \in \{0,1\}$ randomly.
    Step 3: If $S(h,w) = b_1 \oplus b_2$, go to Step 5; else go to Step 4.
    Step 4: Randomly select $p \in \{1,2\}$, and calculate $b_p = \overline{b_p}$.
    Step 5: Randomly rearrange $b_1, b_2$ to $SC_1(h,w), SC_2(h,w)$
    Step 6: Output the 2 shadow images $SC_1$ and $SC_2$
```

Step 4 guarantees $S(h,w) = b_1 \oplus b_2$ in step 3. In step 5, to make all the shadow images be equal to each other, the temporary 2 bits b_1, b_2 are rearranged randomly to corresponding 2 shadow images bits.

In the restoring phase, $S' = SC_1 \otimes SC_2$. If $S(h,w) = 1$, since $S(h,w) = b_1 \oplus b_2$, $SC_1(h,w)$ and $SC_2(h,w)$ are complementary, that is, $SC_1(h,w) = \overline{SC_2(h,w)}$; otherwise, $SC_1(h,w) = SC_2(h,w)$. Therefore, if $S(h,w) = 1$, the restoring result is black. If $S(h,w) = 0$, the restoring result has half chance to be black or white because b_1 is generated randomly.

$S(h,w) = b_1 \oplus b_2$ could be extended to $S(h,w) = b_1 \oplus b_2 \cdots \oplus b_k$ to achieve (k,k) threshold. After encoding the first k bits, the last n - k bits can be specially designed to construct VCETC for (k,n) threshold with comfortable features [32,33].

Based on the above analyses, one RG-based VC for (k,n)-threshold [33] can be derived in Algorithm 2, which will be adopted in our design.

```
Algorithm 2: One typical RG-based VC for $(k,n)$-threshold
    Input: Any binary image $S$ with size of $H \times W$; threshold parameters $(k,n)$
    Output: $n$ binary shadow images $SC_1, SC_2, \cdots SC_n$
    Step 1: For each position $(h,w) \in \{(h,w)|1 \leq h \leq H, 1 \leq w \leq W\}$, repeat Steps 2-4
    Step 2: Set $b_1, b_2, \cdots b_k \in \{0,1\}$ randomly.
    If $S(h,w) \neq b_1 \oplus b_2 \cdots \oplus b_k$, randomly select $p \in \{1,2,\cdots,k\}$, and compute $b_p = \overline{b_p}$.
    Step 3: Compute $N_k = \lfloor n/k \rfloor$ , set $b_{k+1} = b_1, b_{k+2} = b_2, \cdots b_{2k} = b_k, b_{2k+1} = b_1, \cdots b_{N_k \times k} =$
      $b_k$ and $b_{N_k \times k+1} = b_{N_k \times k+2} = b_n = 0$.
    Step 4: Randomly rearrange $b_1, b_2, \cdots b_n$ to $SC_1(h,w), SC_2(h,w), \cdots SC_n(h,w)$
    Step 5: Output $n$ shadow images $SC_1, SC_2, \cdots SC_n$
```

In step 2, the randomness for each encoding process will result in the randomness of each pixel in the shadow image and thus the randomness of the restored secret image, which will be utilized in our designed VCETC to invalid deep learning-based attack. In general, randomness is effective at resisting machine learning.

3. The Designed Method

VC-enhanced text-based captcha (VCETC) architecture of the designed method is exhibited in Figure 4, the algorithmic steps are in Algorithm 3.

```
Algorithm 3: The designed VCETC
    Input: Text content appeared in the captcha; pre-existed traditional captcha generation
           method; VC candidate pool, that is, $\{VC_c, k, n, t\}$, for $c = 1, 2, \cdots, C$
    Output: Output VC-enhanced text-based captcha $S'$
    Step 1: Utilize pre-existed traditional captcha generation method to generate temporary
      captcha, denoted by $S_0$, according to the text content.
    Step 2: Convert $S_0$ into binary image with automatic threshold to obtain $S_1$.
    Step 3: Randomly pick up one VC method from VC candidate pool.Use the VC method to
      encode $S_1$ to obtain $n$ shadow images $SC_1, SC_2, \cdots SC_n$.
    Step 4: Randomly choose $t$ shadow images, denoted by $SC_{i_1}, SC_{i_2}, \cdots SC_{i_t}$, from all the $n$
      shadow images, to restore $S'$ based on superposing $(\otimes)$ operation, where $t \geq k$.
    Step 5: Output VC-enhanced text-based captcha $S'$
```

The basic idea of the designed method is further analyzed as follows:

- In the input, pre-existed traditional captcha generation method is input and based on which we can further improve the performance. Thus, other pre-existed traditional text-based captcha generation method can also be input in our method and our method is only one enhanced method based on pre-existed traditional captcha generation method rather than a redesign.
- In Step 3, the randomness of the selected VC method is applied to the captcha S'. In addition, the random selection of VC method and t further increases the randomness in the captcha S'.
- VC candidate pool, that is, $\{VC_c, k, n, t\}$, for $c = 1, 2, \cdots, C$, can be set up through screening possible VC schemes and their parameters k, n, t whose contrast value is in $[0.14, 0.36]$, where 0.14 is derived from clarity as Figure 3 for human recognition and 0.36 is given by our experiments.
- We can use the VC method to encode the text captcha first and then use pre-existed traditional captcha generation method to proceed the temporary captcha as well.
- In Step 4, we directly stack the selected t shadow images. We can further improve the randomness by performing dynamic stacking from random angles with random velocities like the gif in Reference [34].
- Some other text-based applications can apply our method as well.

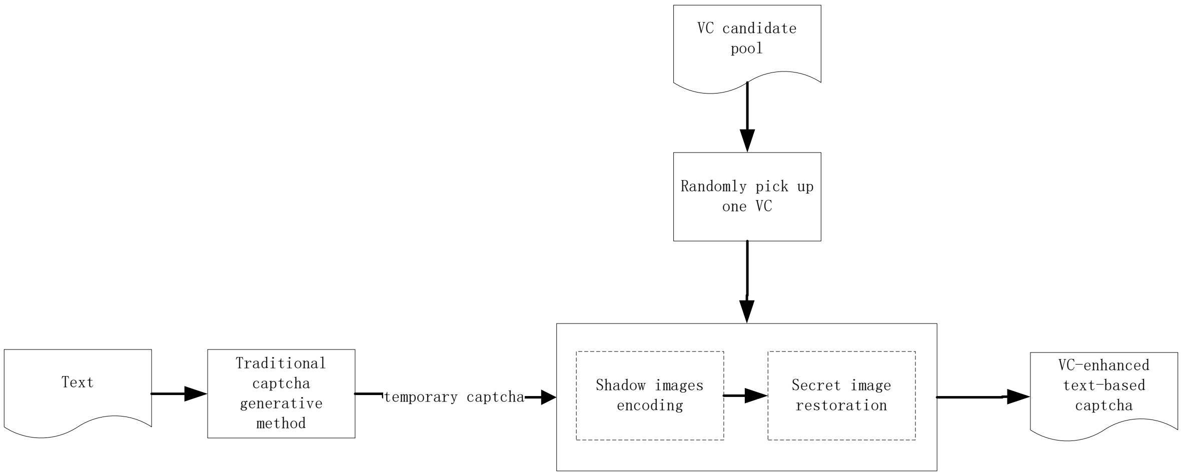


Figure 4. The design concept of our VC-enhanced text-based captcha.

4. Experiments and Comparisons

In this section, experiments are performed to illustrate the effectiveness of the designed method. In the following experiments, we will adopt a traditional typical captcha generation method [6], a VC scheme and deep learning-based breaking methods as Section 2.

To show the effectiveness and advantage of our method, the two CNN-based breaking methods will be separately trained on the captchas generated by the traditional typical method [6] and by our method.

4.1. Breaking Traditional Captcha Generative Captchas by Deep Learning Way

Some captchas generated by traditional captcha generation method are illustrated in Figure 5.

4.1.1. The First Deep Learning Way

The training set contains 100,000 captchas and the testing set contains 10,000 captchas generated by traditional captcha generation method. First, a model, denoted by A_1, is trained with the training set by the first deep learning-based breaking method; then A_1 is utilized to recognize the testing captchas. When applying the deep learning-based breaking method to the above captchas, the success rate is 95%, which indicates that the traditional captchas are easily to be broke by deep learning-based breaking method.

4.1.2. The Second Deep Learning Way

The training set contains 500 captchas and the testing set contains 4540 captchas generated by traditional captcha generation method. First, a model, denoted by A_2, is trained with the training set by the second traditional deep learning-based breaking methods; then A_2 is utilized to recognize the testing captchas. When applying the deep learning-based breaking method to the above captchas, the success rate is 71.52%, which indicates that the traditional captchas are also easily to be broke by deep learning-based breaking method. We note that the second deep learning way only uses 500 captchas in the training process, which is significantly smaller than the first deep learning way.

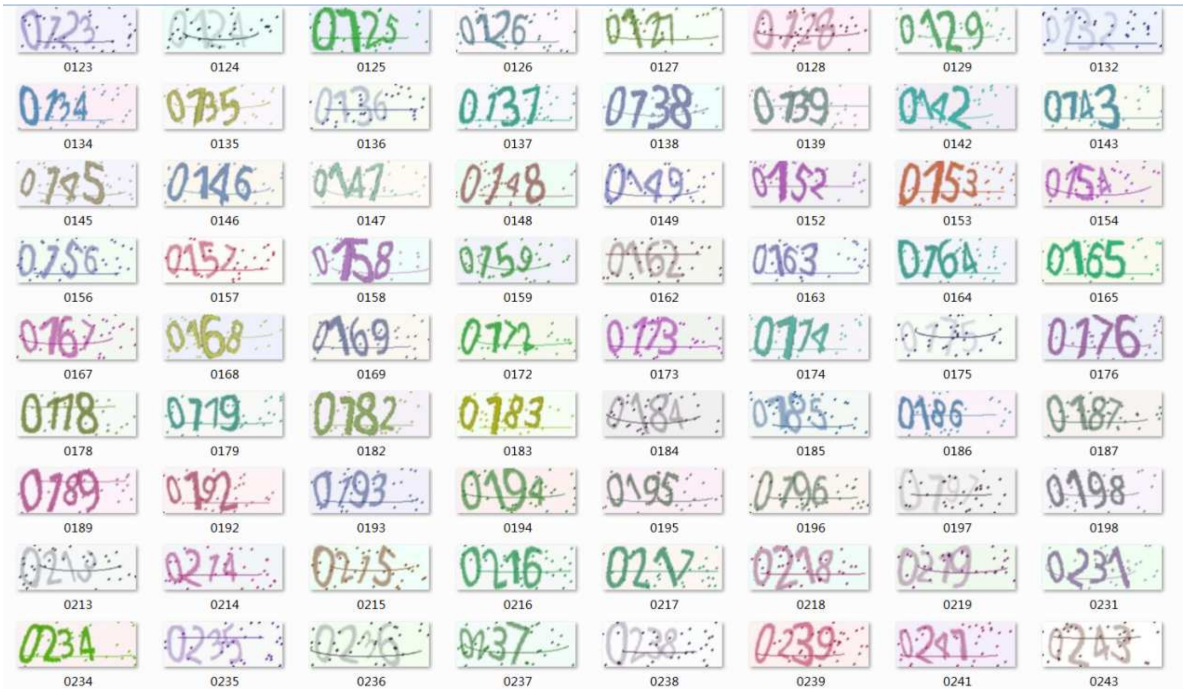


Figure 5. Some captchas in the training set and testing set generated by the traditional text captcha generation method.

4.2. Our Designed Captcha Test by Deep Learning Way

First, we will illustrate the output captchas by our method; second, we will use $A_i, i = 1, 2$ to recognize the testing captchas generated by our method; third, adding captchas generated by our method to the training set to obtain the second model, denoted by $B_i, i = 1, 2$, which will be utilized to recognize the testing captchas generated by our method; finally, the recognized rates with human naked eyes of the traditional captchas and our designed captchas will be evaluated.

Some captchas generated by the proposed method are illustrated in Figure 6, where $k = 2$, $n = 5, t = 3$ and most captchas can be recognized with human naked eyes.

4.2.1. The First Deep Learning Way

We use A_1 to recognize the testing 10,000 captchas generated by our method; Then, adding 8000 captchas generated by our method to the training set to obtain the second model, denoted by B_1, which will be utilized to recognize the testing 2000 captchas generated by our method.

When applying A_1 to the captchas in Figure 6, the success rate is 0.

When adding 8000 captchas generated by our method to the training set to train B_1, the training process is presented in Figure 7. According to Figure 7, the loss function is not decreased all the time and training recognition rate is about 8.7%. Thus it is divergent when VCETCs are added to the training process when we set the motivating rate 80%, which means its training recognition rate is only close to 8.7%.

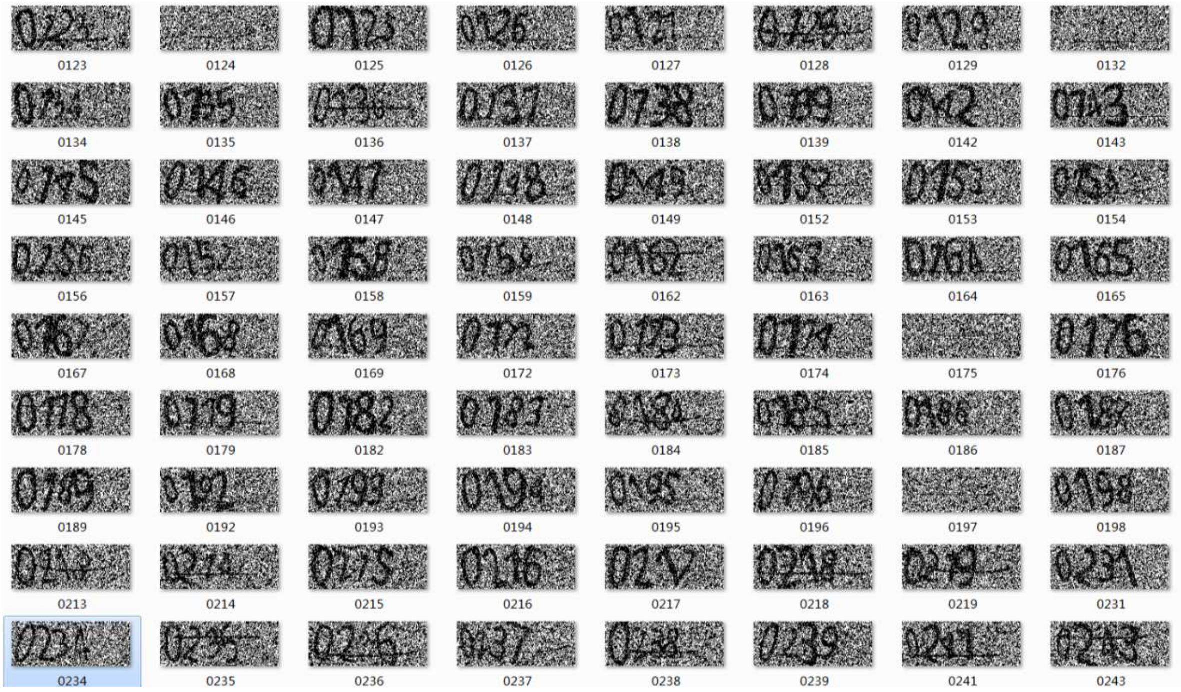
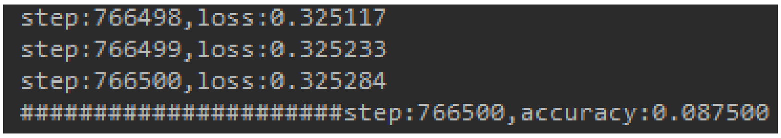


Figure 6. Some captchas generated by the proposed method.



```
step:766498,loss:0.325117
step:766499,loss:0.325233
step:766500,loss:0.325284
######################step:766500,accuracy:0.087500
```

Figure 7. The training process with some captchas generated by the proposed method.

4.2.2. The Second Deep Learning Way

Five hundred captchas generated by our method are used as the training set to obtain the second model, denoted by B_2, which will be utilized to recognize the testing 4540 captchas generated by our method. The success rate to recognize the captchas generated by our method is 53.83%.

4.2.3. The Subjective Recognized Rates with Human Naked Eyes

Finally, the subjective recognized rates with human naked eyes of the traditional captchas and our designed captchas are presented in Table 2. In the testing experiments, we invite totally 20 male volunteers (students aged from 22 to 35 with no experiences of the two types of captchas) to join in the subjective evaluation. Fifty captchas with the same content are randomly picked up from the traditional captchas and our designed captchas, respectively. Every volunteer separately tests totally 50 captchas by himself. After viewing and recognizing each captcha, we will record the recognition situation and recognition speed to obtain Table 2. We find that although the success rate is decreased, it slightly affects user experience.

Table 2. The successfully recognized rates with human naked eyes of the traditional captchas and our designed captchas.

Type	Success Rate	Type	Success Rate	Decreased Value
Traditional captchas	86.75%	Our designed captchas	70.50%	16.25%↓
Traditional captchas within 2 s	83.25%	Our designed captchas within 2 s	63%	20.25%↓

4.2.4. Brief Summary of the Experiments

The brief summary of the experiments is in Table 3. By using our designed VCETC, the recognition rates are in some degree decreased, where the decreasing rates are different when with different deep learning ways. In addition, we present the experimental time complexity of the proposed framework in Table 4. The testing environments are as follows. Windows 7, CPU Intel Core i7, RAM DDR4 16GB, Hard disk 1TB 7200 rpm, matlab 2018a. To generate each captcha, the proposed method needs more 0.0521 (s) than traditional captcha generation method, which is acceptable.

Table 3. Brief summary of the above experiments.

Item	The First Deep Learning Way	The Second Deep Learning Way
Training set	100,000 captchas	500 captchas
Testing set	10,000 captchas	4540 captchas
Success rate of traditional captchas	95%	71.52%
Success rate of our designed captchas	8.7% (the loss function is not decreased all the time with additional 8000 captchas generated by our method)	53.83%

Table 4. The experimental time complexity of the proposed framework (s).

Method	Number of Generated Captchas	Total Generating and Storage Time	Average Generating and Storage Time
Traditional captcha generation method	2500	165	0.0661
The proposed method	2500	295	0.1182

Based on the above experiments we conclude and analyze as follows.

- Due to the features of the randomness for each encoding process in VC and its visual recognizability with naked human eyes, our designed VCETC can in some degree enhance traditional captchas to resist some deep learning-based ways even our designed VCETC are used as the training set.
- Due to the feature of visual recognizability with naked human eyes, VCETCs are suitable for human eyes.
- According to subjective test, our designed VCETC slightly affects user experience with lower storage space, that is, the binary captcha needs a lower storage space and transmission bandwidth than color ones.

4.3. Use-Case Scenario

We will give a use-case scenario of our designed VCETC as follows. For the static text captchas in the original website, just like Figure 5, we can add our method directly to the generation process of the original text captchas, to generate the enhanced text captchas as shown in Figure 6. In such a way, we can complete the application of our designed VCETC.

5. Conclusions

This paper designed a new visual cryptography (VC)-enhanced text-based captcha (VCETC), where the features of the randomness for each encoding process in VC and its visual recognizability with naked human eyes are utilized to countermeasure the automatic recognition of text-based captchas by some deep learning-based ways. Experiments validate the recognition rate is decreased by using our designed VCETC. The user experience is acceptable for our designed VCETC. In the future, we will (1) test more typical and improved deep learning-based ways, (2) exploit more admirable applied ways of VC to captcha, (3) apply VC to many other fields, such as enhancing text-based identification to resist automatic recognition and keywords-based secret information to resist automatic monitoring.

More specifically, we may further extend our work in the following ways.

- There are many practically oriented programs for solving the captchas problem to circumvention the need of human participation expected by website, which are not based on CNN, such as "Universal Share Downloader" (USD) based on plugins and direct optical character recognition (OCR) to recognize some typical captchas. Due to the features of the randomness for each encoding process in VC, our method may enhance such text captchas.
- VCETCs are applied to image-based captchas [35] to enhance them.
- To further improve our method, we may use recommendation mechanisms to recommend text-based captchas close to user's characteristics and profile [36], and individual differences in cognitive processing [37].
- We will provide additional information and discussion to elaborate more on the use case scenario, and how we envision to include the recommender systems.
- Our method can add many dynamic mechanisms to further improve the performance.
- Other recent attempts to improve text-based captchas have been proposed in the scientific literature as well. We will compare our method to the more state of the art enhanced methods.

Author Contributions: Data curation, F.L.; Formal analysis, W.Q.Y.; Investigation, Y.L.; Methodology, X.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (Grant Number: 61602491) and the Key Program of the National University of Defense Technology (Grant Number: ZK-17-02-07).

Acknowledgments: The authors would like to thank the editor and the anonymous reviewers for their valuable comments. The authors would like to thank Guixin Ye and Zhanyong Tang for their kind discussions, supports and valuable experiments.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ahn, L.V. Telling humans and computers apart automatically. *Commun. ACM* **2004**, *47*, 56–60. [CrossRef]
2. Chow, Y.W.; Susilo, W.; Thorncharoensri, P. CAPTCHA Design and Security Issues. In *Advances in Cyber Security: Principles, Techniques, and Applications*; Li, K.C., Chen, X., Susilo, W., Eds.; Springer: Singapore, 2019; pp. 69–92._4. [CrossRef]
3. Bursztein, E.; Aigrain, J.; Moscicki, A.; Mitchell, J.C. The End is Nigh: Generic Solving of Text-based CAPTCHAs. In Proceedings of the Usenix Conference on Offensive Technologies, San Diego, CA, USA, 19 August 2014.
4. George, D.; Lehrach, W.; Kansky, K.; Lázaro-Gredilla, M.; Laan, C.; Marthi, B.; Lou, X.; Meng, Z.; Liu, Y.; Wang, H. A generative vision model that trains with high data efficiency and breaks text-based CAPTCHAs. *Science* **2017**, *358*, eaag2612. [CrossRef] [PubMed]
5. Ye, G.; Tang, Z.; Fang, D.; Zhu, Z.; Feng, Y.; Xu, P.; Chen, X.; Wang, Z. Yet Another Text Captcha Solver: A Generative Adversarial Network Based Approach. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, ON, Canada, 15–19 October 2018; ACM: New York, NY, USA, 2018; pp. 332–348. [CrossRef]
6. Tengxing Breaking Captchas. 2017. Available online: https://github.com/tengxing/tensorflow-learn/tree/master/captcha (accessed on 15 Nov 2017).
7. Lin, G.; Shen, W. Research on convolutional neural network based on improved Relu piecewise activation function. *Procedia Comput. Sci.* **2018**, *131*, 977–984. [CrossRef]
8. Iliev, A.; Kyurkchiev, N.; Markov, S. On the approximation of the step function by some sigmoid functions. **2017**, *133*, 223–234. [CrossRef]
9. Lecun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
10. Von Ahn, L.; Maurer, B.; McMillen, C.; Abraham, D.; Blum, M. reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science* **2008**, *321*, 1465–1468. [CrossRef] [PubMed]
11. Saadat Beheshti, S.M.R.; Liatsis, P.; Rajarajan, M. A CAPTCHA model based on visual psychophysics: Using the brain to distinguish between human users and automated computer bots. *Comput. Secur.* **2017**, *70*, 596–617. [CrossRef]
12. Kim, S.; Choi, S. *DotCHA: A 3D Text-Based Scatter-Type CAPTCHA*; Web Engineering; Bakaev, M., Frasincar, F., Ko, I.Y., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 238–252.
13. Naor, M.; Shamir, A. Visual cryptography. In *Advances in Cryptology-EUROCRYPT'94 Lecture Notes in Computer Science, Workshop on the Theory and Application of Cryptographic Techniques, Perugia, Italy, 9–12 May 1994*; Springer: Berlin/Heidelberg, Germany, 1995; pp. 1–12.
14. Weir, J.; Yan, W. A comprehensive study of visual cryptography. In *Transactions on DHMS V, LNCS 6010*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 70–105.
15. Wang, Z.; Arce, G.R.; Di Crescenzo, G. Halftone visual cryptography via error diffusion. *IEEE Trans. Inf. Forensci. Secur.* **2009**, *4*, 383–396. [CrossRef]
16. Li, P.; Liu, Z.; Yang, C.N. A construction method of (t,k,n)-essential secret image sharing scheme. *Signal Process. Image Commun.* **2018**, *65*, 210–220. [CrossRef]
17. Kafri, O.; Keren, E. Encryption of pictures and shapes by random grids. *Opt. Lett.* **1987**, *12*, 377–379. [CrossRef]
18. Yang, C.N.; Wu, C.C.; Wang, D.S. A discussion on the relationship between probabilistic visual cryptography and random grid. *Inf. Sci.* **2014**, *278*, 141–173. [CrossRef]
19. Yan, X.; Liu, X.; Yang, C.N. An enhanced threshold visual secret sharing based on random grids. *J. Real-Time Image Process.* **2018**, *14*, 61–73. [CrossRef]
20. Wang, W.; Liu, F.; Guo, T.; Ren, Y. Temporal Integration Based Visual Cryptography Scheme and Its Application. In Proceedings of the Digital Forensics and Watermarking: 16th International Workshop, IWDW 2017, Magdeburg, Germany, 23–25 August 2017; pp. 406–419.
21. Li, Y.; Guo, L. Robust Image Fingerprinting via Distortion-Resistant Sparse Coding. *IEEE Signal Process. Lett.* **2018**, *25*, 140–144. [CrossRef]

22. Chavan, P.V.; Atique, M.; Malik, L. Signature based authentication using contrast enhanced hierarchical visual cryptography. In Proceedings of the Electrical, Electronics and Computer Science, Bhopal, India, 1–2 March 2014; pp. 1–5.
23. Luo, H.; Lu, Z.M.; Pan, J.S. Multiple Watermarking in Visual Cryptography. In *Proceedings of the International Workshop on Digital Watermarking*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 60–70.
24. El-Latif, A.A.A.; Abd-El-Atty, B.; Hossain, M.S.; Rahman, M.A.; Alamri, A.; Gupta, B.B. Efficient quantum information hiding for remote medical image sharing. *IEEE Access* **2018**. [CrossRef]
25. Wang, G.; Liu, F.; Yan, W.Q. Basic Visual Cryptography Using Braille. *Int. J. Digit. Crime Forensics* **2016**, *8*, 85–93. [CrossRef]
26. Cheng, Y.; Fu, Z.; Yu, B.; Shen, G. A new two-level QR code with visual cryptography scheme. *Multimedia Tools Appl.* **2017**. [CrossRef]
27. Komargodski, I.; Naor, M.; Yogev, E. Secret-Sharing for NP. *J. Cryptol.* **2017**, *30*, 444–469, [1403.5698]. [CrossRef]
28. Belazi, A.; El-Latif, A.A.A. A simple yet efficient S-box method based on chaotic sine map. *Opt. Int. J. Light Electron Opt.* **2017**, *130*, 1438 – 1444. [CrossRef]
29. Liu, Y.; Yang, C.; Wang, Y.; Zhu, L.; Ji, W. Cheating identifiable secret sharing scheme using symmetric bivariate polynomial. *Inf. Sci.* **2018**, *453*, 21–29. [CrossRef]
30. Liu, F.; Wu, C.; Qian, L. Improving the visual quality of size invariant visual cryptography scheme. *J. Vis. Commun. Image Represent.* **2012**, *23*, 331–342. [CrossRef]
31. Yan, X.; Lu, Y.; Huang, H.; Liu, L.; Wan, S. Clarity Corresponding to Contrast in Visual Cryptography. In Proceedings of the Social Computing: Second International Conference of Young Computer Scientists, Engineers and Educators, ICYCSEE 2016, Harbin, China, 20–22 August 2016; pp. 249–257._23. [CrossRef]
32. Yan, X.; Wang, S.; Niu, X. Threshold construction from specific cases in visual cryptography without the pixel expansion. *Signal Process.* **2014**, *105*, 389–398. [CrossRef]
33. Yan, X.; Lu, Y.; Liu, L.; Wan, S. Random Grids-Based Threshold Visual Secret Sharing with Improved Visual Quality. In Proceedings of the Digital Forensics and Watermarking: 15th International Workshop, IWDW 2016, Beijing, China, 17–19 September 2016; pp. 209–222._16. [CrossRef]
34. Visual cryptography in Wikipedia. 2019. Available online: https://en.wikipedia.org/wiki/Visual_cryptography (accessed on 15 November 2017).
35. Alqahtani, F.H.; Alsulaiman, F.A. Is image-based CAPTCHA secure against attacks based on machine learning? An experimental study. *Comput. Secur.* **2020**, *88*, 101635. [CrossRef]
36. Pouli, V.; Kafetzoglou, S.; Tsiropoulou, E.E.; Dimitriou, A.; Papavassiliou, S. Personalized multimedia content retrieval through relevance feedback techniques for enhanced user experience. In Proceedings of the 2015 13th International Conference on Telecommunications (ConTEL), Graz, Austria, 13–15 July 2015; pp. 1–8. [CrossRef]
37. Belk, M.; Germanakos, P.; Fidas, C.; Holzinger, A.; Samaras, G. Towards the Personalization of CAPTCHA Mechanisms Based on Individual Differences in Cognitive Processing. In *Human Factors in Computing and Informatics*; Holzinger, A., Ziefle, M., Hitz, M., Debevc, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2013; pp. 409–426.