LETTER

CAPTCHA Image Generation Systems Using Generative Adversarial Networks*

Hyun KWON[†], ***Nonmember***, **Yongchul KIM**[††], ***Member***, **Hyunsoo YOON**[†], ***Nonmember***, ***and*** **Daeseon CHOI**[†††a)], ***Member***

SUMMARY We propose new CAPTCHA image generation systems by using generative adversarial network (GAN) techniques to strengthen against CAPTCHA solvers. To verify whether a user is human, CAPTCHA images are widely used on the web industry today. We introduce two different systems for generating CAPTCHA images, namely, the distance GAN (D-GAN) and composite GAN (C-GAN). The D-GAN adds distance values to the original CAPTCHA images to generate new ones, and the C-GAN generates a CAPTCHA image by composing multiple source images. To evaluate the performance of the proposed schemes, we used the CAPTCHA breaker software as CAPTCHA solver. Then, we compared the resistance of the original source images and the generated CAPTCHA images against the CAPTCHA solver. The results show that the proposed schemes improve the resistance to the CAPTCHA solver by over 67.1% and 89.8% depending on the system.

key words: *CAPTCHA, generative adversarial network, deep convolutional network*

1. Introduction

Internet technology and mobile computing systems are becoming pervasive and enabling users to be connected to web services at any time. However, this convenient accessibility has also allowed malicious attackers to intrude into web services. Especially, distributed denial-of-service and massive spam are the well-known autonomous attacks. To prevent this type of attacks, CAPTCHA, the completely automated public Turing test to tell computers and humans apart, can be used in computing applications to determine whether a user is human. CAPTCHA images containing text should be distorted randomly so that only humans can identify [1]. When the distortion and tilting of the characters are significant, even a human may not be able to recognize the characters. Thus, it is important to generate CAPTCHA images that only humans can easily decipher. Nevertheless, there are several studies [2] aiming to solve CAPTCHA, such as optical character recognition and DeCAPTCHA software. The general idea of these schemes is to select highly correlated characters from either a database or mapping table after dividing CAPTCHA characters into pixel segments [3], [4]. However, these CAPTCHA solving schemes strongly depend on the information in the database or mapping table. Moreover, distortions above 5% in the characters from a mapping table result in a high probability of misidentification of the CAPTCHA image [5], [6].

Generative adversarial networks (GANs) have been recently introduced as a generative model for machine learning [7]. Using a GAN, a generator can create data that a discriminator is not able to distinguish from a general dataset. In particular, a deep convolutional GAN [8] scheme learns from some images to generate altered ones. By adopting this GAN technique into a CAPTCHA image generation system, it can create an image that a machine cannot recognize but a human can easily decipher. In this paper, we introduce two CAPTCHA image generation systems by using GAN-based techniques. The contribution of this paper is threefold. First, to the best of our knowledge, this is the first study that presents GAN-based CAPTCHA image generation systems that either add distance values to the original images or compose multiple source images. Second, we introduce a new GAN architecture that includes a distance parameter to generate various CAPTCHA images by adjusting this parameter. Third, we verify the performance of the proposed systems using CAPTCHA images from various commercial websites and a CAPTCHA solver.

The rest of this paper is structured as follows: in Sect. 2, we introduce the proposed CAPTCHA generation systems. Performance results and analysis of the proposed systems are presented in Sect. 3. Section 4 concludes the paper.

2. Proposed Method

In general, our systems consist of input CAPTCHA images that after passing through a GAN generates modified images. We propose two GAN schemes, namely, distance GAN (D-GAN) and composite GAN (C-GAN). The D-GAN scheme uses distance values from the original CAPTCHA images to prevent a CAPTCHA solver from recognizing the images. The C-GAN scheme applies a com-

Manuscript received August 8, 2017.
Manuscript revised October 13, 2017.
Manuscript publicized October 26, 2017.

[†]The authors are with School of Computing, Korea Advanced Institute of Science and Technology, Korea.
[††]The author is with Electrical and Computer Engineering, North Carolina State University, USA.
[†††]The author is with Dept. of Medical Information, Kongju National University, Korea.

*This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2016-0-00173, Security Technologies for Financial Fraud Prevention on Fintech) and the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2016R1A4A1011761).

a) E-mail: sunchoi@kongju.ac.kr

DOI: 10.1587/transinf.2017EDL8175

position method into a GAN process to improve resistance against a CAPTCHA solver. A GAN is an unsupervised machine learning process where generator and discriminator neural networks compete to produce zero-sum outcomes. The generator provides an output value to the discriminator whenever it receives an input value. Then, the discriminator calculates the difference between the received value from the generator and the original source data by using a loss function. To minimize this loss function, the generator adjusts its parameters and provides a different output value to the discriminator. As this process iterates, the output data from the generator approach to an original source data. In this process, the main objective of the discriminator is to distinguish between the output of the generator and the original source data. In contrast, the objective of the generator is to create output data as close to the original source data as possible. Let us denote with G and D the generative and discriminative models, respectively. The models play the following two-player minimax game with value function $V(G, D)$ as described in [7]:

$$\begin{aligned}&\min_G \max_D V(D, G)\\ &= E_{x\sim p_{data}(x)}[\log D(x)] + E_{z\sim p_z(z)}[\log(1 - D(G(z)))] \quad (1)\end{aligned}$$

where $D(x)$ and $G(z)$ represent discriminative and generative model functions, respectively. This objective function results in the same fixed point of the dynamics of G and D, but provides much stronger gradients at the early learning stage. That is, the distribution of z approaches the distribution of x as the learning process evolves. For example, if we input a CAPTCHA image into a GAN process as source data, the output image from the generator will become close to this image as the process iterates. Figure 1 shows an example of a CAPTCHA image resulting from a GAN generator at different iterations.

2.1 D-GAN Scheme

The proposed D-GAN scheme adds distance values to the original CAPTCHA images to generate new CAPTCHA images. According to the distance values, a CAPTCHA solver would not be able to recognize the generated CAPTCHA images. However, it is also possible that the human cannot recognize the generated CAPTCHA images for large distance values. Therefore, an appropriate limit for the distance values should experimentally be determined. Figure 2 shows the proposed D-GAN architecture. The generator provides a function $G(z)$ where z is a noise. The distribution of $G(z)$ is approaches that of x by minimizing the loss function from the discriminator. The loss function can be expressed as

$$loss\ function = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}). \quad (2)$$

where y and $\hat{y}$ represent a valid probability distribution and an unscaled log probability, respectively. The proposed D-GAN modifies this loss function by adding distance value $Dist(C, z)$ that can be adjusted by constant value C. The modified loss function can be expressed as

$$\begin{aligned}&loss\ function^* = loss\ function + Dist_{z\sim p_z(z)}(C, z)\\ &= -y \log \hat{y} - (1 - y) \log(1 - \hat{y}) + y \log\left(1 + \frac{1 - C}{C(1 - \hat{y})}\right). \quad (3)\end{aligned}$$

Through the modified loss function, a slightly altered CAPTCHA image is generated. When constant value C is close to 1, the generated CAPTCHA image is similar to the original. If C is greater than 1, the difference between the generated and the original images will be substantial. Figure 3 shows CAPTCHA images generated by the D-GAN for different C values. As an alternative to the D-GAN, it is possible to add noise directly to the original image to create a simple CAPTCHA image, i.e., every pixel of the original image will have an equal amount of noise value. However, it is more likely that the CAPTCHA solver can easily recognize the noise added simple CAPTCHA images compared to D-GAN generated images. The detailed process of generating a D-GAN CAPTCHA image is described in Algorithm 1.

2.2 C-GAN Scheme

The proposed C-GAN scheme is based on a composing technique to generate a single CAPTCHA image and improve resistance against CAPTCHA solvers. Figure 4 shows

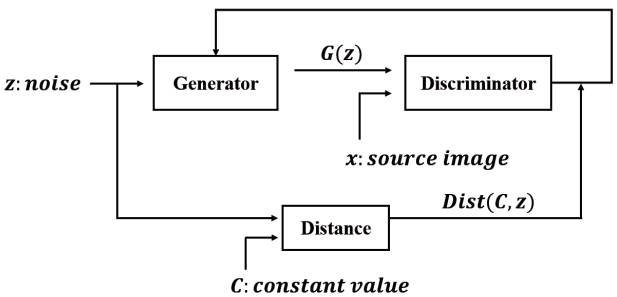
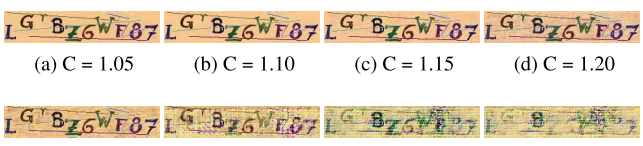


Fig. 2 Proposed D-GAN architecture.

(a) 100 cycles (b) 200 cycles (c) 300 cycles (d) 400 cycles

Fig. 1 CAPTCHA images produced by a GAN generator at different iteration cycles.



(a) C = 1.05 (b) C = 1.10 (c) C = 1.15 (d) C = 1.20

(e) C = 3 (f) C = 6 (g) C = 9 (h) C = 12

Fig. 3 CAPTCHA images generated by the D-GAN for different C values.

Algorithm 1 D-GAN process

Input: noise sample z ~ $p_g(z)$, sample x ~ $p_{data}(x)$, constant value C, number of iterations n, and number of discriminator's steps j.

Process:

for n step **do**

for j step **do**

Update the discriminator by maximizing gradient $\max_D[\log D(x)] + [\log(1 - D(G(z)))] + Dist(C, z)$

end for

Update the generator by minimzing gradient $\min_G[\log(1 - D(G(z)))]$

end for

(a) 100 iterations (b) 300 iterations

(c) 500 iterations (d) 700 iterations

Fig. 4 C-GAN learning process using two source CAPTCHA images in a single batch.

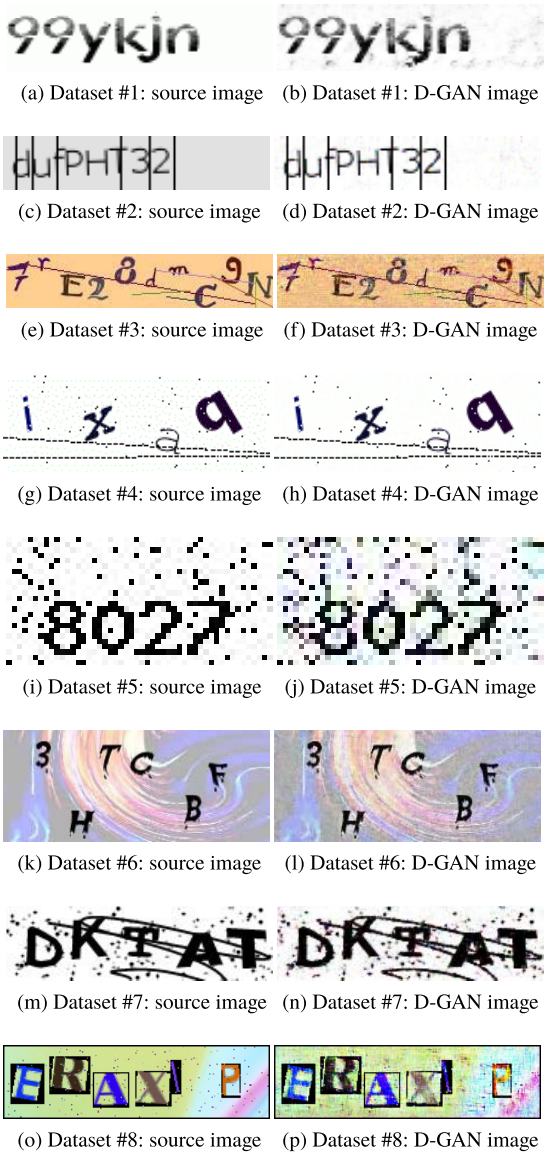
(a) Source images (b) Combined image

Fig. 5 Combination of CAPTCHA images to generate composite CAPTCHA image in Fig. 4.

an example of a C-GAN learning process using two source CAPTCHA images included in a single batch (three out of the four images are the same). As the iterations increase, the original CAPTCHA images continuously change. The final C-GAN CAPTCHA image created by composing two images from the example is shown in Fig. 5. The generated image can notably increase the resistance against a CAPTCHA solver as shown in the following section. The enhancement obtained from the C-GAN CAPTCHA generation system is given by the combination of multiple CAPTCHA images in a same batch during the learning process. More precisely, some of the pixels are either deleted or replaced with other pixels. Thus, a C-GAN CAPTCHA image contains information from various object in multiple CAPTCHA images.

3. Experiments and Analysis

In this section, we show experimental results and analyze the performance of the proposed D-GAN and C-GAN schemes by evaluating the recognition rates of the GSA Captcha breaker [9], one of the famous CAPTCHA solvers. We used 8 different datasets CAPTCHA images from commercial websites and 100 CAPTCHA images per dataset. We used TensorFlow [10], one of the most widely used open source libararies for machine learning. Figure 6 shows



(a) Dataset #1: source image (b) Dataset #1: D-GAN image

(c) Dataset #2: source image (d) Dataset #2: D-GAN image

(e) Dataset #3: source image (f) Dataset #3: D-GAN image

(g) Dataset #4: source image (h) Dataset #4: D-GAN image

(i) Dataset #5: source image (j) Dataset #5: D-GAN image

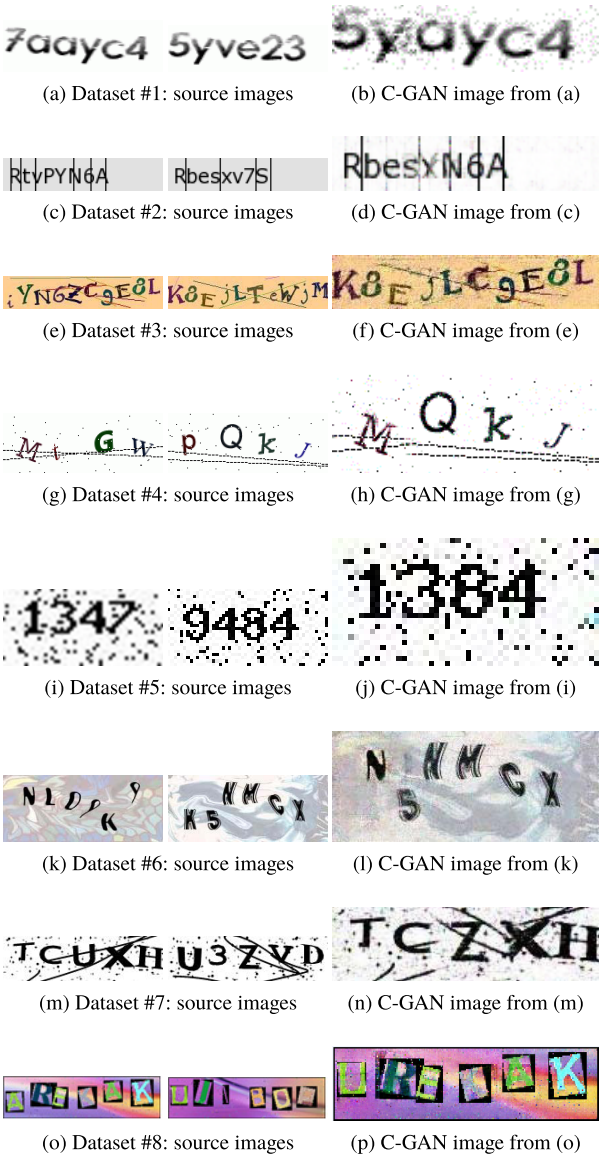
(k) Dataset #6: source image (l) Dataset #6: D-GAN image

(m) Dataset #7: source image (n) Dataset #7: D-GAN image

(o) Dataset #8: source image (p) Dataset #8: D-GAN image

Fig. 6 Source images and D-GAN images with C = 1.2.

examples of the original CAPTCHA images and D-GAN-generated images for the 8 styles. Examples of the C-GAN-generated CAPTCHA images are shown in Fig. 7. The recognition rates of the CAPTCHA solver for the source and generated images are listed in Table 1. For the original CAPTCHA images, the recognition rates are highly variables because each dataset has different characteristics such as text distribution, distortion level, character rotation, and background image. For the D-GAN CAPTCHA images, the recognition rates substantially decrease compared to those of the original images. Furthermore, the rates further decrease when the *C* value increases. However, the rate reduction is also variable due to the different background images of each dataset. For the C-GAN CAPTCHA images, the resistance against the CAPTCHA solver is further improved given the composing technique. In particular, the recognition rates for datasets #1, #2, and #8 are 0% because the



(a) Dataset #1: source images (b) C-GAN image from (a)

(c) Dataset #2: source images (d) C-GAN image from (c)

(e) Dataset #3: source images (f) C-GAN image from (e)

(g) Dataset #4: source images (h) C-GAN image from (g)

(i) Dataset #5: source images (j) C-GAN image from (i)

(k) Dataset #6: source images (l) C-GAN image from (k)

(m) Dataset #7: source images (n) C-GAN image from (m)

(o) Dataset #8: source images (p) C-GAN image from (o)

Fig. 7 C-GAN images generated from two sources.

Table 1 CAPTCHA solver recognition rates.

Type	Source	D-GAN		C-GAN
		C = 1.05	C = 1.2	
Data #1	39%	30%	28%	0%
Data #2	24%	14%	8%	0%
Data #3	59%	7%	6%	4%
Data #4	16%	12%	9%	8%
Data #5	60%	11%	2%	2%
Data #6	4%	3%	2%	3%
Data #7	19%	8%	6%	6%
Data #8	6%	4%	0%	0%

background images in those dataset are clear compared with those of the other datasets. Thus, it is likely that the background images are replaced with peripheral images during the composition process.

4. Conclusion

In this paper, we propose GAN-based CAPTCHA image generation systems to enhance CAPTCHA images. The D-GAN system uses distance values during the learning process to generate new CAPTCHA images that a CAPTCHA solver cannot recognize. The CAPTCHA image is further enhanced by using the C-GAN system based on a composing technique. The C-GAN system can also be used to increase the quantity of images when the dataset is limited. To evaluate the performance of the proposed schemes, we compared the recognition rates of the CAPTCHA solver for the original CAPTCHA images and those generated with the proposed schemes. The results show that the proposed schemes significantly improve the CAPTCHA image performance. Future work will focus on creating CAPTCHA images without using source images. Recently, new CAPTCHA systems with natural images are introduced in the real world. Soon after, it is shown that the natural images can be easily recognized by using neural networks. Applying our proposed scheme into the natural image systems will be an interesting research topic in our future work.

References

[1] L. Von Ahn, M. Blum, N.J. Hopper, and J. Langford, "CAPTCHA: Using hard AI problems for security," International Conference on the Theory and Applications of Cryptographic Techniques, Lecture Notes in Computer Science, vol.2656, pp.294–311, Springer, 2003.

[2] G. Moy, N. Jones, C. Harkless, and R. Potter, "Distortion estimation techniques in solving visual CAPTCHAs," Proc. 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, pp.23–28, 2004.

[3] A. Hindle, M.W. Godfrey, and R.C. Holt, "Reverse engineering CAPTCHAs," 15th Working Conference on Reverse Engineering, WCRE '08, pp.59–68, 2008.

[4] E. Bursztein, J. Aigrain, A. Moscicki, and J.C. Mitchell, "The end is nigh: Generic solving of text-based captchas," WOOT, 2014.

[5] B. Biggio, I. Corona, D. Maiorca, B. Nelson, N. Šrndić, P. Laskov, G. Giacinto, and F. Roli, "Evasion attacks against machine learning at test time," Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science, vol.8190, pp.387–402, Springer, 2013.

[6] N. Papernot, P. McDaniel, S. Jha, M. Fredrikson, Z.B. Celik, and A. Swami, "The limitations of deep learning in adversarial settings," 2016 IEEE European Symposium on Security and Privacy (EuroS&P), pp.372–387, 2016.

[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," Advances in Neural Information Processing Systems, pp.2672–2680, 2014.

[8] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv preprint arXiv:1511.06434, 2015.

[9] "GSA captcha breaker," GSA - Softwareentwicklung und Analytik GmbH, Available: https://captcha-breaker.gsa-online.de/, 2017.

[10] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G.S. Corrado, A. Davis, J. Dean, M. Devin, et al., "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," arXiv preprint arXiv:1603.04467, 2016.