VRC-GraphNet: A Graph Neural Network-Based Reasoning Framework for Attacking Visual Reasoning Captchas

Botao Xu and Haizhou Wang(✉)

School of Cyber Science and Engineering, Sichuan University, Chengdu 610065, China
2020141530039@stu.scu.edu.cn, whzh.nc@scu.edu.cn

Abstract. Captchas are widely used to distinguish between human and machine programs and protect computers from malicious attacks. However, with the development of image recognition and deep learning techniques, the attack success rate of traditional text-based and image-based captcha is getting higher. This leads to increasing demand for more secure captcha. In recent years, some captcha service providers such as Tencent, NetEase, Geetest, etc. put forward novel visual reasoning captchas to improve the safety level, and reduce the risk of attacks. There has been little research on this kind of novel captcha. Existing method mainly uses modular method to break it, but has to train separately and is still insufficient for reasoning task. In order to solve above challenges for visual reasoning captcha, this paper introduces a novel end-to-end graph reasoning network to crack the visual reasoning captcha for the first time. We use object detection model to identify all the objects in the captcha. Then, we extract the distribution of question attention and image features to build a graph neural network. Finally, an end-to-end reasoning framework for attacking visual reasoning captcha is constructed by using reasoning module to integrate multi-modal. We achieve a higher success rate of attack on some very popular visual reasoning captchas. The results will provide technical and theory support for the security evaluation of captchas, and promote research on more secure captchas.

Keywords: captchas · visual reasoning · object detection · graph neural network · features extraction

1 Introduction

1.1 Background

With the rapid development of information technology, network security has become an increasing concern. To defend malicious attacks or automated bots, various websites and mobile applications often employ captchas. Captcha is usually composed of a series of digits, letters, images, audio or other information

H. Duan et al. (Eds.): SecureComm 2023, LNICST 567, pp. 165–184, 2025.
https://doi.org/10.1007/978-3-031-64948-6_9

that is challenging for automated programs to recognize. Each type of captcha requires users to solve a unique task that is easy for human to complete, yet difficult for machines and programs. At present, common captchas include text-based captchas, image-based captchas, voice recognition captchas, slide-based captchas and so on [29]. To improve the security of captchas, developers constantly added anti-recognition mechanisms into them, such as background noise, character rotation, overlap, distortion [2,5,7]. However, with the rapid development of character recognition [21], image recognition [10], these captchas can be easily broken by attackers using machine learning methods [3] and deep learning methods [15,23,30,31]. Meanwhile, the captchas are increasingly difficult to be recognized by human because of the complex anti-recognition mechanisms deployed by captcha developers for improving security. However, the user experience is greatly reduced and the security does not significantly improve [1].

To solve above problems, the company Tencent proposed a visual reasoning captcha called Visual Turing Test (VTT)[1] [25] for the first time. This kind of captcha gives an image containing many objects, and a question. Users must choose the correct object according to the question and click on the specific area to pass the captcha. Subsequently, to improve the security and user experience of captchas, other popular captcha service providers, such as Geetest[2], Netease[3], Dingxiang[4], Shumei[5], and Xiaodun[6] proposed similar visual reasoning captchas to defend against robot programs. The questions of these captcha challenges usually include the investigation of object attributes, such as shape, color, size. Some questions also include complex visual and spatial logic relations, such as relative position, relative size, the same color and shape.

Since the captcha has only been proposed in the last few years, few researches have focused on the security of this kind of captcha. Among them, the designer of VTT evaluated its security by carrying out a relation network attack test, but only achieved a 4.7% success rate [25], indicating that this type of captcha has a good defense effect.

Visual reasoning captchas are required to solve problems similar to Visual Question Answer (VQA) [12,19]. However, with the rapid development of deep learning technology, visual reasoning captchas are facing increasingly enormous security threats. Wang et al. [27] were first to carry out the study of VTT attack using modular method [14]. To the best of our knowledge, this is the only attacking work on visual reasoning captchas. However, the ability of their model to learn object attributes and relative relation is still insufficient. Some potential logical relations will be ignored when different modules are used to filter objects directly. In addition, the results of former module will greatly affect the latter

[1] https://007.qq.com/online.html.
[2] https://www.geetest.com/show.
[3] https://dun.163.com/trial/space-inference.
[4] https://www.dingxiang-inc.com/business/captcha.
[5] https://www.ishumei.com/trial/captcha.html.
[6] https://sec.xiaodun.com/onlineExperience/spatialReasoningSelection.

modules in their method. For example, if the results of detection module are not good, this will affect the final reasoning results of integration module.

1.2 Challenges

At present, the research on attacking visual reasoning captchas mainly faces the following three challenges:

The first challenge is that there is little research on visual reasoning captcha at present [27], and existing research doesn't provide public available dataset. Moreover, many visual reasoning captcha service providers add anti-crawler mechanism in their websites, which makes it very difficult to obtain and construct large-scale and high-quality datasets.

The second challenge is how to integrate the multimodal features of visual reasoning captchas for training. To solve the task of visual reasoning captcha, attackers need to learn semantic information and image information at the same time, find the target object according to the logical relation, and locate the specific target region. However, the existing VQA work [12,19] lacks the reasoning ability of complex logical relation for this type of captcha.

The third challenge is that previous attack on visual reasoning captcha schemes [27] lacks end-to-end logical reasoning, requiring separate training and module assembly. Moreover, procedure parser needs to be designed for each type of captcha, which requires a large labor cost.

1.3 Contributions

For the above challenges, this paper proposes a general, graph neural network-based end-to-end reasoning framework, which can crack existing popular visual reasoning captchas with a high attack success rate. Specifically, we built a web crawler and successfully collected large scale of visual reasoning captchas from multiple popular captcha service providers. The attack framework is composed of object detection module, question encoding module, image feature extraction module and graph reasoning module. First, we use the object detection method based on Mask R-CNN [9] to identify all objects in the image. Then we extract the distribution of question attention and object image features as multi-modal. Finally, we use each detected object as a node and relative relations as edges to construct a multi-step graph reasoning network. By reasoning with multi-modal, we get the predict answer. Meanwhile, our model can achieve a relevant high attack success rate with a small training number of captchas. The main contributions of this paper are summarized as follows:

- **To the best of our knowledge, we are the first to propose an end-to-end reasoning framework for attacking visual reasoning captchas.** The framework is based on graph neural network, which adopts GRU-like updating mechanism to spread the visual and position information in the network. It integrates multimodal features to carry out successful attacks on multiple very popular visual reasoning captchas. The success attack rate of

VTT, Geetest, NetEase, Shumei, Xiaodun are 89.5%, 76.8%, 72.0%, 100.0% and 89.7%, respectively.

- **In view of the difficulty of logical reasoning in visual reasoning captchas, we propose a novel method of extracting absolute position, relative position, visual and relative visual features of objects in visual reasoning captchas for the first time.** We use these features to learn the visual and spatial position relationships of objects, complete relevant reasoning tasks, and visualize the reasoning process. The results show that by using these features, our model has good performance in answering relevant logical reasoning questions.

2 Related Work

2.1 Visual Question Answering

The VQA is a research field that integrates image understanding, natural language processing and machine learning. Recent research models mainly include modular methods and end-to-end methods. On the one hand, models such as NS-VQA [28], XNMs [22] adopt modular methods that transform questions into procedures which can filter objects with irrelevant attributes. By constructing modules, they used these procedures to get the final text answers. On the other hand, model of MAC [13], FiLM [20] used end-to-end approach, which integrates semantic and image features to achieve good results.

However, existing VQA models simply extract semantic and image features to fuse multi-modal, which lack the ability to learn logical relation in visual reasoning captchas. At the same time, the key to crack the visual reasoning captcha is to output accurate answer location, while most of existing VQA models output text answers. Therefore, to solve complex logic reasoning, special model for visual reasoning captcha is needed.

2.2 Security Analysis of Captchas

The existing research on the security of captchas can be divided into text-based captchas [5,7,24], image-based captchas [30], audio and video-based captchas [18], slider-based captchas [29,30] and visual reasoning captchas [25]. We will focus on three types of popular captchas including text, image, visual reasoning captchas to introduce the related cracking and defense work as follows.

Text-Based Captchas. As the first proposed captcha [24], text-based captcha requires the user to give the correct character sequence in order according to the text image. Existing text-based captchas adopt multiple anti-recognition mechanisms: Amazon uses rotated characters [7], Google ReCaptcha uses distorted characters [7], to increase the difficulty of cracking them. Other captchas increase the attack difficulty by employing complex character structure, such as Microsoft which uses two-layer characters [5] and Apple which uses overlapping

characters [26]. Instead of changing the character shape or structure, Sina and Scihub [2] add noise such as dots, lines and shadows into the background to interfere with text recognition. It is important to note that current text-based captchas increase the cracking difficulty by using more than one anti-recognition mechanism.

Although text-based captcha defense mechanism keeps evolving, it doesn't hinder the development of text recognition technology from cracking it. There has been a lot of works on how to crack the various text-based captchas. Gao et al. [6] successfully cracked hollow captchas using color fill segmentation algorithm. In view of the noise in text-based captcha, Chen et [2] proposed a variety of noise removal methods based on spatial domain filter, Gibbs and Hough transform, and morphology to break anti-recognition mechanism. Dionysiou et al. [3] systematically summarized the work of attacking text-based captchas using machine learning and deep learning, and points out that text-based captcha is no longer secure.

In general, attack methods against text-based captcha are getting more effective with the development of artificial intelligence. At present, the improvement space of text-based captcha defense has become very limited, and the security is not enough to resist the existing state of the art methods.

Image-Based Captchas. Existing image-based captchas can be divided into select-based captchas, slider-based captchas and click-based captchas [29,30]. Compared with text-based captchas, they are more friendly and have richer information in images, promoting a wide range of applications.

Select-based captchas require users to select images of a specified category from a group of images according to object hint, such as Asirra [4] which requires users to select cat images from 12 images. Zhao et al. [30] achieved 83.25% attack success rates on select-based ReCaptcha v2 using image detection and classification models. Slider-based captchas [29,30] require users to drag the slider to the specified position, and distinguish human from machine according to the accuracy of the mouse trajectory and the position of the slider. The process of cracking slider-based captchas is mainly divided into two steps. First, the slider and target position are obtained. Then, a slide track is made according to the offset distance of the target position. A script is used to simulate the drag of mouse. Although slider-based captchas are no longer secure enough, they are still widely deployed due to low cost and user friendliness. For click-based captchas [29,30], users need to click characters in sequence following the hint, which can be regarded as a simplification of text-based captchas. The task behind click-based captchas is the same as text-based captchas, with similar anti-recognition mechanisms. The security of click-based captchas is also decreasing because of excellent character recognition technology.

Visual Reasoning Captchas. Visual reasoning captcha is a new kind of captcha and has appeared in recent years. At present, only Wang's research team [27] has worked on visual reasoning captchas. In order to crack the visual reasoning

captchas, they proposed a model composed of semantic parsing, detection, classification and integration modules. Adopting the work from [14], the semantic parsing module parses the input question and converts it into a reasoning program. The detection module uses Faster R-CNN [8] to locate the object and obtain simple attributes, such as color, size and shape. The classification module uses SENet [11] to further classify the subtle attributes including tilt direction and character category. Finally, the integration module uses reasoning programs to filter out redundant objects, and the final object left is the predicted answer.

Wang's work has achieved a high success rate, but there are still some problems to be solved. (1) With the increasing and upgrading of visual reasoning captchas, new measures have been taken to make the number of objects to be detected increase and logical relationships more complex. In Wang's method, the detection module may fail to recognize objects that have occlusion, which mean an object is partially blocked by other objects. Part of logical relation reasoning is unable to carry out because occluded objects are unable to participate in reasoning. (2) There are many object categories in captchas, which makes it harder for detection and classification module to learn the image features extracted by the network. As a result, the programs used in reasoning are very likely to filter out objects relevant to the question and miss some underlying logical relationship. (3) Their method needs to train separately and assemble modules, which requires a large labor and computation cost.

In order to solve above problems, this paper proposes an end-to-end reasoning framework for attacking visual reasoning captchas. To address visual reasoning captchas that have more objects in images and richer logical information in questions, we use graph neural network as model foundation due to the stronger ability in learning relationship between objects. Moreover, in view of more object categories in visual reasoning captchas, we adopt Mask R-CNN in our detection module to make sure all objects are detected as many as possible. By extracting the object features and question attention distribution of the captchas, the model is capable of learning multi-modal features. Through the updating mechanism of GRUs, the model propagates and fuses multi-modal features in the network and successfully attacks multiple visual reasoning captcha schemes. The end-to-end reasoning does not need to train modules separately, which decreases our labor cost.

3 Methodology

In this section, we describe in detail our framework used to attack visual reasoning captcha. As shown in Fig. 1, our framework consists of data collection module, object detection module, question encoding module, image feature extraction module and graph reasoning module.

3.1 Data Collection Module

At present, there are few studies on visual reasoning captchas. Only Wang et al. [27] have carried out relevant attack research. But they did not make their

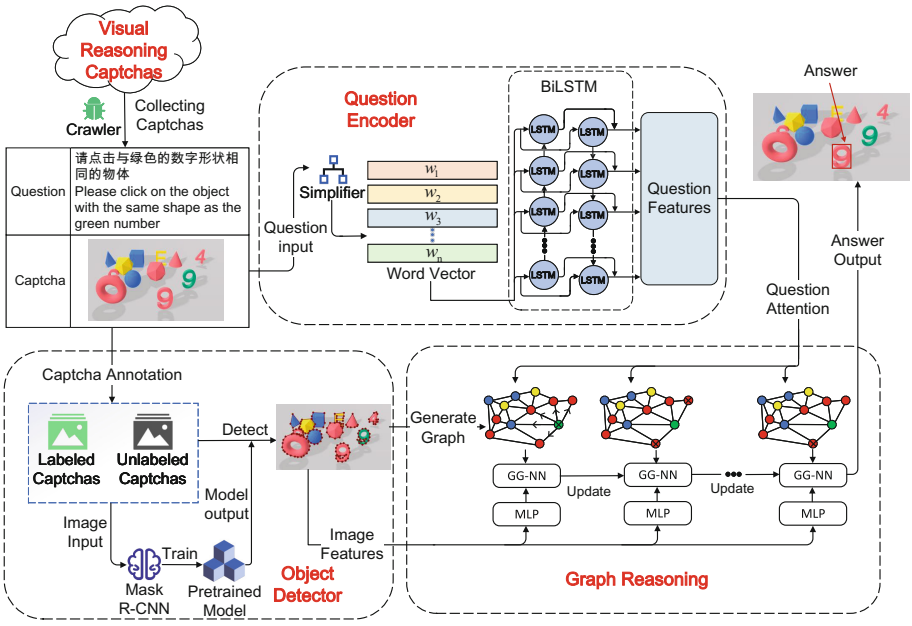


Fig. 1. Framework for attacking visual reasoning captchas

dataset public, which has greatly limited the research work in this field. To collect data for experiments, we developed a web crawler program for visual reasoning captchas platforms, and collected a large number of visual reasoning captchas in the form of image-question pairs. Subsequently, dataset construction and data annotation are carried out.

Visual reasoning captcha service providers usually adopt strict anti-crawl means to avoid large-scale malicious collection of captchas, which limits the dataset construction in this research field. In order to bypass anti-crawl mechanism and collect sufficient visual reasoning captchas, we develop customized web crawlers using the function library Selenium in Python, which can control browser to dynamically click elements in websites. Specifically, we collected images of visual reasoning captchas and corresponding questions, and adopted a de-duplication mechanism to ensure that the captchas collected were not repetitive.

We use crawlers to obtain visual reasoning captchas of VTT, NetEase, Geetest, and Shumei. As the official website of Xiaodun provides few captchas, we tried our best to collect, but collected less than 10,000 captchas. So we developed a program which generates new questions based on existing captcha images to expand the size of Xiaodun captcha dataset. Finally, we collected a large number of captchas from VTT, Geetest, NetEase, Xiaodun, Shumei, with 10,000, 10,000, 10,000, 10,000, and 300 captchas, respectively. Note that captcha data was collected from February 26, 2023 to April 1, 2023.

3.2 Object Detection Module

After data collection, we use object detection module to process visual reasoning captcha images and obtain locations of all objects in the image. The output is object bounding boxes and object labels.

Object Detection. Our object detection module uses Mask R-CNN [9], a deep learning-based model that can simultaneously detect and segment objects in images. Compared with other target detection models, Mask R-CNN has more accurate positioning and segmentation. It can detect all the categories in captchas, such as uppercase or lowercase letters, digits, geometric objects, while significantly reducing the cost and time of our detection. Therefore, this paper chooses it as object detection model.

Manually Labeling and Object Detection Training. In order to use Mask R-CNN for object detection, we select a number of 250, 50, 150, 100, 25 images from VTT, Geetest, NetEase, Xiaodun, Shumei, respectively and manually label them. Subsequently, we trained Mask R-CNN using Resnet-50 network parameters pre-trained on ImageNet, and generated separate model parameters for different visual reasoning captchas. Finally, the object bounding boxes and labels detected in each captcha are obtained. We made a preliminary attempt to ensure that all objects in the captchas are detected, and discovered that a threshold of score higher than 0.7 can achieve best results. Therefore, we set the threshold for object detection.

After the training of object detection is completed, we count the number of object categories and calculate the detection accuracy of each kind of captcha. Among them, Tencent VTT includes three categories of letters, numbers, and geometric objects, a total of 53 types of objects. NetEase includes three categories of letters, numbers, and geometric objects, a total of 57 types of objects. Xiaodun includes letters, numbers, geometric objects, with a total of 46 types of objects. The captchas of Geetest and Shumei only include the category of geometric objects, and there are 5 types and 7 types of objects respectively.

3.3 Question Encoding Module

Our question encoding module consists of a simplifier and a BiLSTM network. This paper innovatively proposes a simplifier algorithm for questions of visual reasoning captchas. The simplifier refers to preprocessing the question input into the BiLSTM, removing and replacing redundant strings. By doing this, the number of question vocabulary is reduced and the complexity of subsequent reasoning network is decreased. For example, a question in VTT '请点击正对你的字母' (Please click the letter facing you), after processing by our simplifier, the new question is '正对的字母' (the facing letter), which reduces unnecessary strings '请点击' (Please click) and '你' (you). We also remove character punctuation in the question. Some simplifier examples are shown in Table 1.

Table 1. Examples of questions before and after the simplification process.

Platform	Origin Question	Simplified Question
VTT	请点击侧对着你的字母“t” Please click the letter “t” that is side facing you	侧对着的字母t the side facing letter t
Geetest	请点击在大型黄色立方体左方的黄色物体。 Please click the yellow object to the left of the large yellow cube.	在大黄色正方体左侧的黄色物体 the yellow object left of the large yellow cube
NetEase	请点击小写w朝向一样的大写B Please click the lowercase w with the same direction as uppercase B	小写w朝向一样的大写B the lowercase w with the same direction as uppercase B
Shumei	点击图中最小的绿色长方体 Click the smallest green rectangular in the image	最小的绿色长方体 the smallest green cube rectangular
Xiaodun	请点击与倾斜的物体形状相同的物体 Please click the object that has the same shape as the tilted object	与倾斜的物体形状相同的物体 the object that has the same shape as the tilted object

After being processed by the simplifier, we establish vocabulary dictionary based on all the questions in each type of captcha. We use Jieba[7], a function library in Python which can segment Chinese sentences, to establish a vocabulary-to-number mapping table. Subsequently, we map the word vectors $<w_1, w_2, ..., w_t>$ to number vectors $<n_1, n_2, ..., n_t>$. After preprocessed by the word embedding layer and the multi-layer perceptron, the word embedding e is input into the BiLSTM network. Since the questions of visual reasoning captchas in Shumei and Netease are not complex, the size of the word embedding layer is set to 64, and the word embedding used by the rest of the captchas is set to 128. The size of the hidden layer of the BiLSTM network is 256, and the number of layers is 2. We extract the output of the BiLSTM network as the semantic information representation. After that, we linearly transform it and use SoftMax function to standardize. The word embeddings are combined to use the self-attention network to calculate the attention distribution weight of each feature w_{feat}. The question attention distribution is divided into absolute position, relative position, visual feature, and relative visual feature attention according to image features.

3.4 Image Feature Extraction Module

In this module, we analyze and extract image features of visual reasoning captchas. These features are classified into four categories: absolute position feature, relative position feature, visual feature, and relative visual feature.

[7] https://pypi.org/project/jieba/.

Visual Feature. Existing works on CLEVR [13,20] show that using the feature extracted by the conv4 layer in the pre-trained Resnet101 network can learn the visual attributes of objects well. Different from these works which extracted the features of the entire image, we segment the captcha into object images according to the object bounding boxes returned by the object detection module. Then, we extract the image feature f_{vis} of each object image.

Absolute Position Feature. We receive the object bounding boxes $bbox = [x, y, w, h]$ from the output of object detection module, where x, y denotes the upper-left coordinates of the object bounding box, and w, h denotes the width and height of the object bounding box. According to previous work [16], the position and relative position features of objects can well represent the relationship among objects, and also satisfy the logical reasoning requirements for positional relationships in visual reasoning captchas. Therefore, we define $f_{abs\_pos} = [\frac{x}{W}, \frac{y}{H}, \frac{x+w}{W}, \frac{y+h}{H}, \frac{w \cdot h}{W \cdot H}]$, where W, H represent the width and height of the captcha, respectively. Our results show that the first four features can learn the position of an object in the captcha. The last feature $\frac{w \cdot h}{W \cdot H}$ represents the size of an object in the captcha, therefore can answer questions about the size of objects.

Relative Position Feature. Some questions require users to accurately describe the position relation among objects in visual reasoning captchas, so we extracted the position relationship feature f_{edge} between each two objects in the captcha. To describe the relative position, we use the polar coordinates proposed in [19]. Specifically, We adopt polar coordinates to better represent the relative relation among objects. The c_x, c_y represent the center point of an object, θ represents the angle relation and ρ represents the distance relation:

$$\theta = \frac{arctan(\frac{c_{y_j} - c_{y_i}}{c_{x_j} - c_{x_i}})}{\frac{\pi}{2}} \tag{1}$$

$$\rho = \frac{\sqrt{(c_{x_j} - c_{x_i})^2 + (c_{y_j} - c_{y_i})^2}}{\sqrt{W^2 + H^2}} \tag{2}$$

$$f_{edge} = [\theta, \rho], \tag{3}$$

We define $f_{rel\_pos} = [f_{edge}, f_{vis_j}, f_{abs\_pos_j}]$, which represents the relative position feature between object i and object j.

Visual Relative Feature. Visual relative feature is used to deal with questions about the same color or shape. We define $f_{obj_n} = [f_{vis_n}, f_{abs\_pos}, l_n]$ as all visual features of the object n, where l_n is the object label value output from the object detection module. The $f_{rel\_vis} = [f_{obj_i}, f_{obj_j}]$ is used to represent the relative visual feature between object i and object j.

3.5 Graph Reasoning Module

We build a graph reasoning module, which receives the object bounding boxes output from the former object detection module, the weight of the question attention distribution output from the question encoding module and four kinds of image features from image feature extraction. We construct the corresponding graph neural network so as to represent object relation in the captcha. After multi-step reasoning, the network finally outputs the predicted object bounding box as the answer.

Graph Neural Network Construction. We use object detection module to get all the objects in the captcha. These objects are defined as nodes $V = \{v_i\}_{i=1}^{N}$ of the graph neural network. The relationship between each two objects is defined as edges $E = \{e_{i,j}\}_{i,j=1}^{N}$ of the graph neural network. By doing this, the graph neural network $G = \{V,\ E\}$ is constructed. Node attributes contain extracted absolute position feature f_{abs_pos} and visual feature f_{vis}, and edge attributes contain extracted relative position feature f_{rel_pos} and relative visual feature f_{rel_vis}.

Multi-step Reasoning. After constructing the graph neural network, we first use the multi-layer perceptron to process the image features. Then, inspired by GG-NN [17], we add a GRU-like updating mechanism into the graph network and implement a multi-step reasoning process:

$$f_{mm} = W_{mm}(W_{ques} atten_{rel} + W_{img} f_{rel}) \tag{4}$$

$$A_{rel}^{n} = tanh(f_{mm} h_{i-1}^{rel,n-1}) \tag{5}$$

$$h_i^{rel,n} = GRU(A_{rel}, h_{i-1}^{rel,n}), \tag{6}$$

In (4) (5) (6), the rel includes relative position feature and relative visual feature. The W_{ques}, W_{img}, W_{mm} represent the fully connected layer parameters of question attention, image feature and multi-modality, respectively. The f_{mm} represents the multi-modal feature after integration, and $h_i^{rel,n}$ represents the hidden layer state of the ith object after n steps reasoning. After completing the multi-step reasoning, we use the last hidden layer state of the captcha to get the final answer prediction score:

$$score = \sum tanh(atten) \cdot tanh(h^{\{rel,abs\}}), \tag{7}$$

We regard the final problem as a multi-classification task, which means selecting the object with the highest probability among the N objects in a captcha. The probability of the object i is expressed by $p_i = \frac{e^{score_i}}{\sum_{j=1}^{N} e^{score_j}}$. Cross entropy is used as the loss function of our model:

$$L = -\sum_{i=1}^{N} c_i \cdot log(p_i), \tag{8}$$

where c_i is 1 when the object is the answer of the visual reasoning captchas, and 0 when it is not.

Table 2. Information of visual reasoning captchas from different platforms.

Platform	Question attribute	Class number	Example label
VTT	color	4	red, yellow, blue, green
	shape	53	cube, sphere, uppercase N, lowercase a, number 5
	size	2	biggest, smallest
	direction	2	facing, side facing
	location	2	right below, above
Geetest	color	4	red, yellow, blue, green
	shape	5	cube, sphere, cone, cylinder, polyhedron
	size	2	biggest, smallest
	location	2	in front, behind, left, right
NetEase	color	4	red, yellow, blue, green
	shape	57	cylinder, number 6, uppercase D, lowercase y
	direction	2	facing, side facing
Shumei	color	4	red, yellow, blue, green
	shape	7	cylinder, rectangular, hexagonal prism
	size	1	smallest
Xiaodun	color	4	red, yellow, blue, green
	shape	46	ring, polyhedron, uppercase A, number 7
	size	2	biggest, smallest
	direction	2	tilt, non-tilt
	location	1	above

4 Experiments

In this section, we evaluate the performance of the proposed model framework by attacking various visual reasoning captchas. First, we describe the experiment setup in our work. After this, we conduct a couple of experiments to verify the superiority of graph reasoning model in solving visual reasoning captchas.

4.1 Experiment Setup

In our work, the experiment of object detection module is on Intel(R) Xeon(R) E5-2680 v4 CPU and TITAN X-12G GPU, 15 GB memory. The rest of the experiments are conducted on 12th Gen Intel(R) Core(TM) i7-12700H CP and NVIDIA GeForce RTX 3060 GPU, with 16 GB of RAM. All deep learning models are implemented using Pytorch v1.7.0.

4.2 Analysis of Visual Reasoning Captchas

To evaluate the performance of the proposed graph reasoning model, we carry out a detailed analysis of visual reasoning captchas. We collect relevant information including question attributes, class number and example label from different captcha platforms, as shown in Table 2. From the table, we can discover

that question attributes of all the visual reasoning captchas include color and shape. The VTT, Geetest, and Xiaodun are more difficult, which include the question attribute such as size, position, direction, upper or lower case of letter. Besides, they have more types of shapes, which demonstrates that breaking visual reasoning captchas is not easy.

4.3 Train and Test Attack on Visual Reasoning Captchas

In this experiment, we use our model to conduct training on various visual reasoning captcha datasets. The VTT, Geetest, NetEase and Xiaodun's train set, validation set and test set are 8,000, 1,000 and 1,000 question-image pairs, respectively. Shumei's train set, validation set and test set are 240, 30 and 30 question-image pairs, respectively. In this way, the training set comprised 80% of the data, while the validation set and test set each accounted for 10%. The attack success rates on various visual reasoning captchas are shown in Fig. 2.

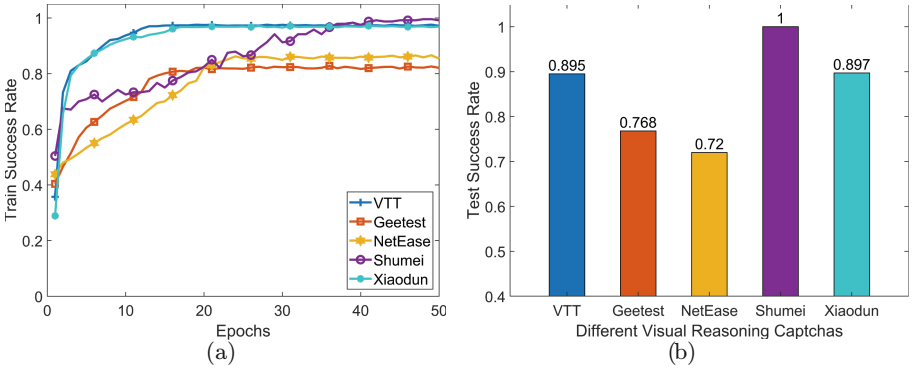


Fig. 2. Train and test attack success rate of different captchas.

As shown in Fig. 2, the training on VTT and Xiaodun reaches convergence around 15 epochs, while Geetest, NetEase and Shumei reach convergence around 35 epochs. The final attack success rate of our model on the test datasets of VTT, Geetest, NetEase, Shumei, Xiaodun are 89.5%, 76.8%, 72.0%, 100.0% and 89.7%, respectively, all of which are more than 70%, indicating that our reasoning model can achieve excellent performance in attacking various multiple visual reasoning captchas tasks. Our model achieves best on Shumei, this may due to the reason that Shumei has simpler geometry objects. Our model does not perform well on Geetest and NetEase, we think that the reason comes down to the complex logic reasoning behind their questions.

4.4 Influence of the Scale of the Dataset

In our work, VTT, Geetest, NetEase, Xiaodun captchas trainsets contain 8,000 pairs of captcha-questions. To explore the performance of our model with a

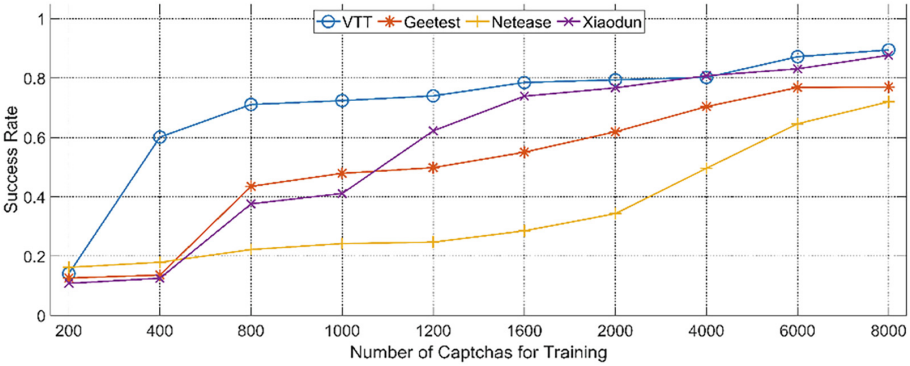
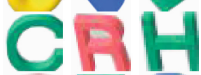


Fig. 3. Success rate of visual reasoning captchas with different training numbers.

small trainset, we train our model with different numbers of captchas. Because the number of Shumei visual reasoning captchas is not enough, the experiment is not carried out on it. The success rate of attacking visual reasoning captchas

Table 3. Proportion and attack success rate of different shapes.

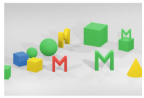
Platform	Shape	Examples	Proportion	ASR
VTT	Geometry		16.5%	76.16%
	Letter		70.9%	90.48%
	Number		12.6%	92.12%
Geetest	Geometry		100.0 %	76.90%
NetEase	Geometry		4.1%	68.59%
	Letter		79.6%	67.48%
	Number		16.3%	90.24%
Shumei	Geometry		100.0%	100.00%
Xiaodun	Geometry		63.9%	92.18%
	Letter		26.6%	78.95%
	Number		9.5%	82.11%

with various scale of trainset is shown in Fig. 3. From the figure, we can see that attack success rates of four kinds of captchas are low when the scale of trainset is 200. But the attack success rate of VTT improves greatly as the scale of trainset reaches 400, which reaches a success rate of around 60% and then slowly increases. The attack success rate of Geetest and NetEase gradually increases with the increasing scale of trainset, but is slower compared to VTT. For Xiaodun, the attack success rate exceeds 50% when the scale of trainset is only 1,200. After 1,200, the attack success rate of it slowly increases. It shows that the success rates of model attacks on all the captchas can exceed 50% when the scale of the trainset is only half of the original dataset. It is proved that our model can also be applied to a small-scale dataset.

4.5 Attack Success Rate of Different Shapes

In this experiment, we have also calculated the attack success rate of different shapes in each visual reasoning captcha scheme, as shown in Table 3. From the table, we can see that our model achieves the highest attack success rates on number in VTT, NetEase and Xiaodun, which all exceed 90%. For geometry shape, Shumei achieves higher success rate compared to other captchas. The reason is that Shumei has only geometry objects. Although there are many shapes with different attributes in VTT and Xiaodun, our model still has a

Table 4. Test attack success rate of visual reasoning captchas using different models.

Platform	Example	Ours	LCGN [12]	VQA [19]	Wang et al. [27]*
Shumei		**100.0%**	10.0%	70.0%	95.9%
Xiaodun		**89.7%**	16.6%	20.3%	79.2%
VTT		**89.5%**	10.9%	19.1%	88.0%
Geetest		76.8%	11.6%	18.6%	**90.8%**
NetEase		72.0%	16.7%	25.0%	**86.2%**

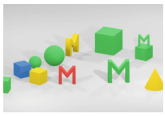
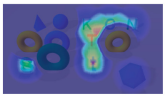
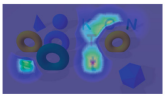
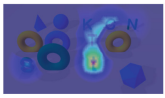
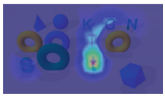
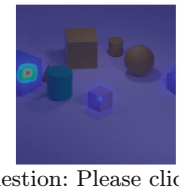
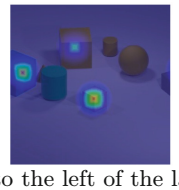
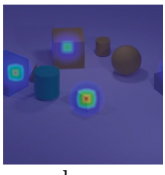
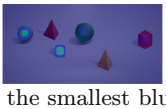
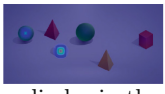
* Note that Wang's work did not make their dataset and code public, and the model is too complex and difficult to reproduce. Therefore, we cite their experimental results directly for comparison.

good performance, which demonstrates that it has the ability to learn multi-modal and the logical reasoning behind the corresponding questions.

4.6 Baseline Studies

In order to evaluate performance, a series of baseline studies are considered in our work. Wang's model [27] does not make the code public, and the model is too complex and difficult to reproduce. So instead of comparing with their model, we select two VQA-related work that can also solve task of visual reasoning captchas. We compare our model with [19], which achieved good results on VQA2.0 dataset and LCGN [12] which achieved excellent performance on CLEVR-ref dataset. We used the same object detection module and compare the different reasoning part. The experimental results are shown in Table 4. Our attacking framework can achieve success rates of 100.0%, 89.7%, 89.5%, 76.8% and 72.0% on the test dataset of Shumei, Xiaodun, VTT, Geetest and NetEase, respectively. The results are far superior to the model LCGN [12] and VQA [19]. Compared to these models, we achieve state of the art on Shumei, VTT and Xiaodun. Compared to Wang's results, we did not achieve the best results on

Table 5. The visualization of captcha reasoning.

Origin	S=1	S=2	S=3	S=4
				
VTT-Question: Please click the letter on the cube				
				
Xiaodun-Question: Please click the lowercase of the red letter				
				
Geetest-Question: Please click the small cube to the left of the large sphere.				
				
Netease-Question: Please click the lowercase u with the same color as the lowercase f				
				
Shumei-Question: Click the smallest blue cylinder in the image				

Geetest and NetEase, which may have something to do with the fact that we used different datasets. Note that we cite their experimental results from [27] directly for comparison because they did not make their dataset and code public.

4.7 Reasoning Visualize

In addition, to better demonstrate the reasoning effectiveness of our model on visual reasoning captchas, we visualize the graph attention distribution in the

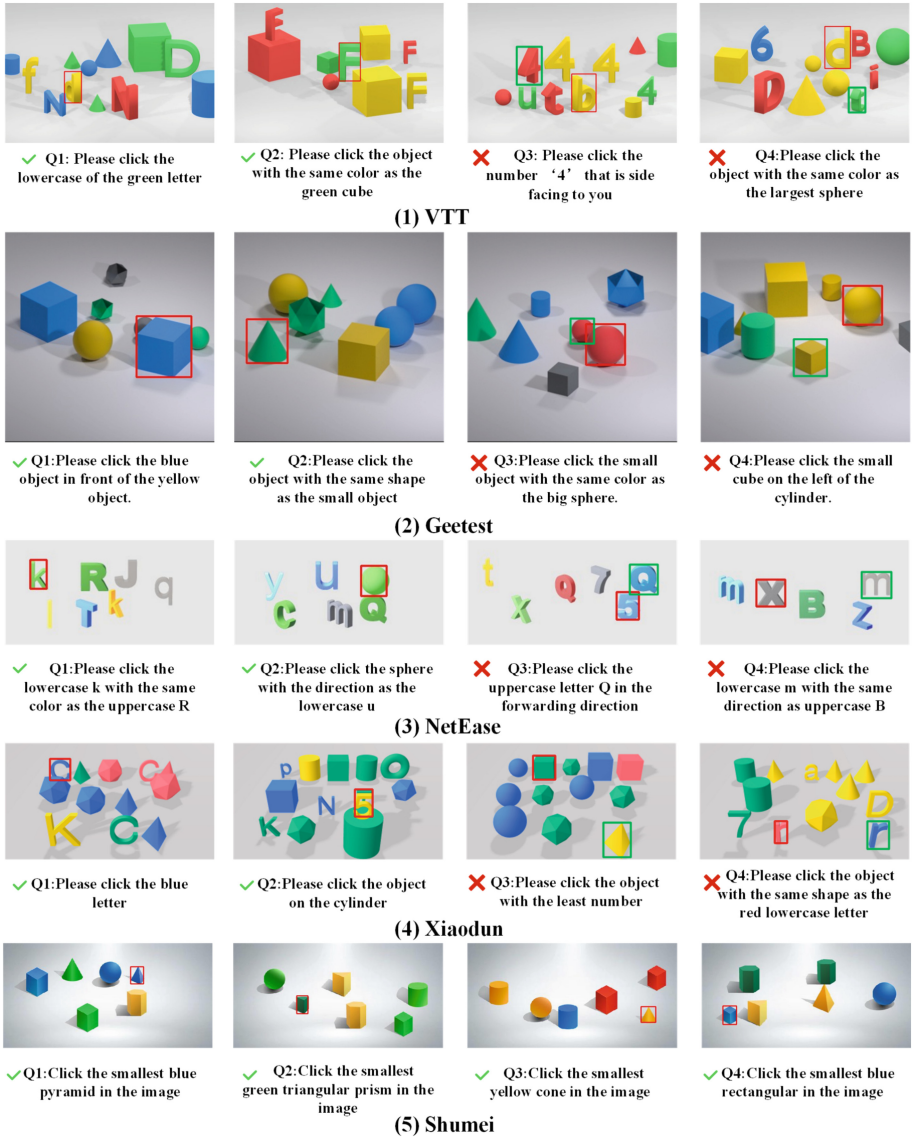


Fig. 4. Successful and failed examples of visual reasoning captchas.

multi-step reasoning process, as shown in Table 5. S represents the Sth step of reasoning. We can learn from the table that the attentive image region changes through out the multi-step reasoning and regions are relevant to the corresponding question. This indicates that our model can complete the logical reasoning in captchas.

4.8 Case Study Experiment

To illustrate the availability of our method, we carried out the case study experiment. Some successful and failed attacks are shown in Fig. 4. Previous experiments show that we can achieve best in Shumei, VTT and Xiaodun. For VTT, we can see that our model detects the side facing objects, but chooses wrong shape in Q3. This failure may due to that the shapes of the two objects are similar. While in Q4 the failure may be the reason that the model gets the wrong size. Although there is only geometry objects in Geetest, the questions include many relative position attributes which make the logical reasoning more complex. The model achieves 100% on Shumei captchas, we think the key reason is that the geometry objects in Shumei are very simple and there are no occlusion. We think that our model still needs improvement in learning the abstract logic behind questions.

5 Conclusion

This paper proposes a novel end-to-end reasoning framework based on graph neural network for attacking visual reasoning captchas. The framework is composed of five modules: data collection module, object detection module, question encoding module, image feature extraction module and graph reasoning module. Specifically, we use the object detection module to extract the object boundary boxes and labels. Based on the absolute position, relative position, visual and relative visual features in captcha images, we integrate the attention distribution output of the question encoding module. We construct a graph neural network with object attributes and relational attributes as nodes and edges, and implement multi-step reasoning by combining the GRU-like updating mechanism. The answer object boundary box of visual reasoning captcha is given. In addition, we trained the model with a small dataset, and the experimental results show that our graph reasoning network can achieve good performance on multiple visual reasoning captchas. We also visualize our reasoning step, which demonstrates that our model has excellent reasoning learning ability. In the future, we will further improve the attack success rate of our model, and we can build a more effective model for transmitting multi-modal, which can be extended to other similar visual reasoning tasks.

Acknowledgements. This work is supported by Development Program of Science and Technology Department of Sichuan Province under grant No. 2023YFG0145 and the National Key Research and Development Program of China under grant No. 2022YFC3303101 and Key Research.

References

1. Bursztein, E., Bethard, S., Fabry, C., Mitchell, J.C., Jurafsky, D.: How good are humans at solving CAPTCHAs? A large scale evaluation. In: Proceedings of the 31st IEEE Symposium on Security and Privacy, pp. 399–413 (2010)
2. Chen, J., Luo, X., Guo, Y., Zhang, Y., Gong, D.: A survey on breaking technique of text-based CAPTCHA. Secur. Commun. Netw. **2017**, 6898617 (2017)
3. Dionysiou, A., Athanasopoulos, E.: SoK: machine vs. machine–a systematic classification of automated machine learning-based CAPTCHA solvers. Comput. Secur. **97**, 101947 (2020)
4. Elson, J., Douceur, J.R., Howell, J., Saul, J.: Asirra: a CAPTCHA that exploits interest-aligned manual image categorization. In: Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS), pp. 366–374 (2007)
5. Gao, H., Tang, M., Liu, Y., Zhang, P., Liu, X.: Research on the security of Microsoft's two-layer captcha. IEEE Trans. Inf. Forensics Secur. **12**(7), 1671–1685 (2017)
6. Gao, H., Wang, W., Qi, J., Wang, X., Liu, X., Yan, J.: The robustness of hollow CAPTCHAs. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer & Communications Security, pp. 1075–1086 (2013)
7. Gao, H.: Robustness of text-based completely automated public Turing test to tell computers and humans apart. IET Inf. Secur. **10**(1), 45–52 (2016)
8. Girshick, R.: Fast R-CNN. In: Proceedings of the 15th IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
9. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the 16th IEEE International Conference on Computer Vision, pp. 2961–2969 (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
11. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition, pp. 7132–7141 (2018)
12. Hu, R., Rohrbach, A., Darrell, T., Saenko, K.: Language-conditioned graph networks for relational reasoning. In: Proceedings of the 17th IEEE International Conference on Computer Vision, pp. 10294–10303 (2019)
13. Hudson, D.A., Manning, C.D.: Compositional attention networks for machine reasoning. arXiv preprint arXiv:1803.03067 (2018)
14. Johnson, J., et al.: Inferring and executing programs for visual reasoning. In: Proceedings of the 16th IEEE International Conference on Computer Vision, pp. 2989–2998 (2017)
15. Li, C., Chen, X., Wang, H., Zhang, Y.: End-to-end attack on text-based CAPTCHAs based on cycle-consistent generative adversarial network. Neurocomputing **443**, 223–236 (2021)
16. Li, L., Gan, Z., Cheng, Y., Liu, J.: Relation-aware graph attention network for visual question answering. In: Proceedings of the 17th IEEE International Conference on Computer Vision, pp. 10313–10322 (2019)
17. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.: Gated graph sequence neural networks. arXiv preprint arXiv:1511.05493 (2015)
18. Meutzner, H., Gupta, S., Kolossa, D.: Constructing secure audio captchas by exploiting differences between humans and machines. In: Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, pp. 2335–2338 (2015)

19. Norcliffe-Brown, W., Vafeias, S., Parisot, S.: Learning conditioned graph structures for interpretable visual question answering. In: Advances in Neural Information Processing Systems, vol. 31 (2018)
20. Perez, E., Strub, F., De Vries, H., Dumoulin, V., Courville, A.: Film: visual reasoning with a general conditioning layer. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, no. 1 (2018)
21. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2016)
22. Shi, J., Zhang, H., Li, J.: Explainable and explicit visual reasoning over scene graphs. In: Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, pp. 8376–8384 (2019)
23. Tang, M., Gao, H., Zhang, Y., Liu, Y., Zhang, P., Wang, P.: Research on deep learning techniques in breaking text-based captchas and designing image-based captcha. IEEE Trans. Inf. Forensics Secur. **13**(10), 2522–2537 (2018)
24. von Ahn, L., Blum, M., Hopper, N.J., Langford, J.: CAPTCHA: using hard AI problems for security. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 294–311. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-39200-9_18
25. Wang, H., Zheng, F., Chen, Z., Lu, Y., Gao, J., Wei, R.: A captcha design based on visual reasoning. In: Proceedings of the 44th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1967–1971 (2018)
26. Wang, P., Gao, H., Guo, X., Xiao, C., Qi, F., Yan, Z.: An experimental investigation of text-based CAPTCHA attacks and their robustness. ACM Comput. Surv. **55**(9), 1–38 (2023)
27. Wang, P., Gao, H., Xiao, C., Guo, X., Gao, Y., Zi, Y.: Extended research on the security of visual reasoning CAPTCHA. IEEE Trans. Dependable Secure Comput. (2023)
28. Yi, K., Wu, J., Gan, C., Torralba, A., Kohli, P., Tenenbaum, J.: Neural-symbolic VQA: disentangling reasoning from vision and language understanding. In: Advances in Neural Information Processing Systems, vol. 31 (2018)
29. Zhang, Y., Gao, H., Pei, G., Luo, S., Chang, G., Cheng, N.: A survey of research on captcha designing and breaking techniques. In: Proceedings of the 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), pp. 75–84 (2019)
30. Zhao, B., et al.: Towards evaluating the security of real-world deployed image captchas. In: Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security, pp. 85–96 (2018)
31. Zi, Y., Gao, H., Cheng, Z., Liu, Y.: An end-to-end attack on text captchas. IEEE Trans. Inf. Forensics Secur. **15**, 753–766 (2019)