Korea Digital Forensics Society 2025 Winter Conference

A Methodology for Automated Resolution of Multi-type CAPTCHA Challenges

JunBeom Cho*, Yunji Park**, Doowon Jeong**

Department of Applied Artificial Intelligence (Bachelor Course), Sungkyunkwan University*
Department of Forensics, Sungkyunkwan University**

This research was supported by Culture, Sports and Tourism R&D Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024
(Project Name: Development of copyright forensic tool technology to identify and detect AI-based copyright infringement content, Project Number: RS-2024-00442014, Contribution Rate: 100%)

Contents

I. Introduction

II. Methodology

III. Detection Stage

IV. Resolution Stage

V. Evaluation

VI. Conclusion

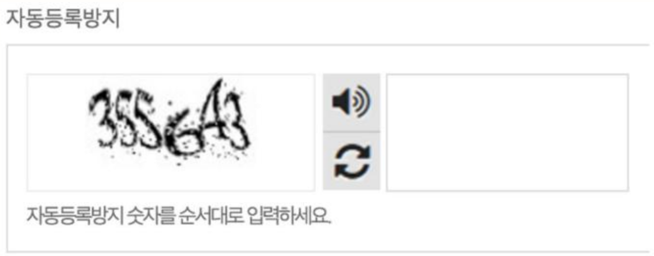
1. Introduction

Background & Problem

- The widespread adoption of the Internet enables large-scale access to global information.
- But also increases cybercrimes such as illegal transactions, account takeovers, and unauthorized content sharing.
- CAPTCHA systems are widely used to block automated access and protect online resources, but advanced ML/DL and multimodal LLMs have reduced their effectiveness.

CAPTCHA (Completely Automated Public Turing test to tell Computers and Human Apart)

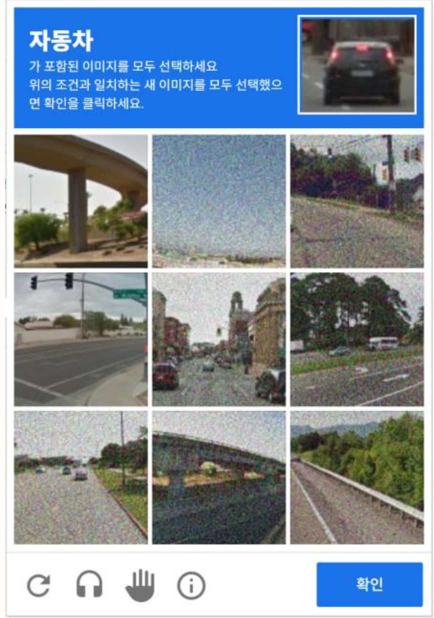
- An authentication technology developed to block automated malicious bot access
- Advances in deep learning have enabled high-accuracy breaking of CAPTCHAs.



자동등록방지

자동등록방지 숫자를 순서대로 입력하세요.

Text based CAPTCHA

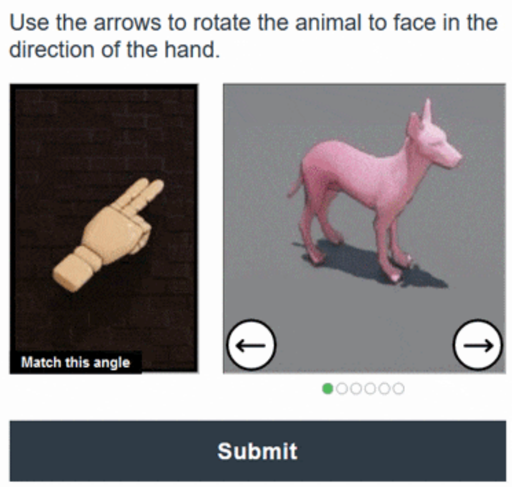


자동차

가 포함된 이미지를 모두 선택하세요

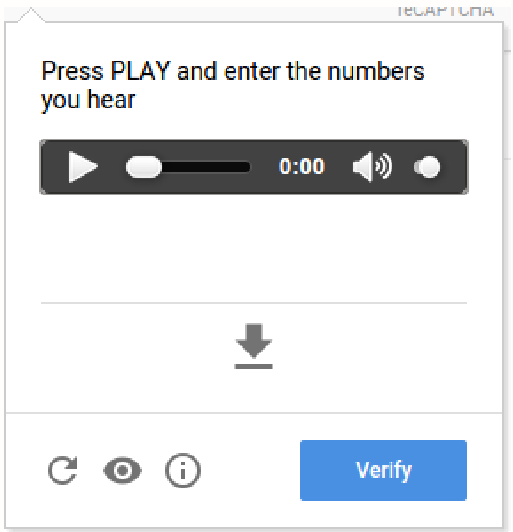
위의 조건과 일치하는 새 이미지를 모두 선택했으면 확인을 클릭하세요.

Image based CAPTCHA



Use the arrows to rotate the animal to face in the direction of the hand.

Reasoning based CAPTCHA



Press PLAY and enter the numbers you hear

Audio / Video based CAPTCHA

Prior Studies

- Image-based CAPTCHAs have also become vulnerable：
  modern object-detection models can successfully decode
  many image type CAPTCHAs.

- Even 3D style CAPTCHAs have shown weaknesses：
  automated preprocessing and segmentation attacks can yield
  high character/overall solve rates, revealing design vulnerabilities.

Prior Studies

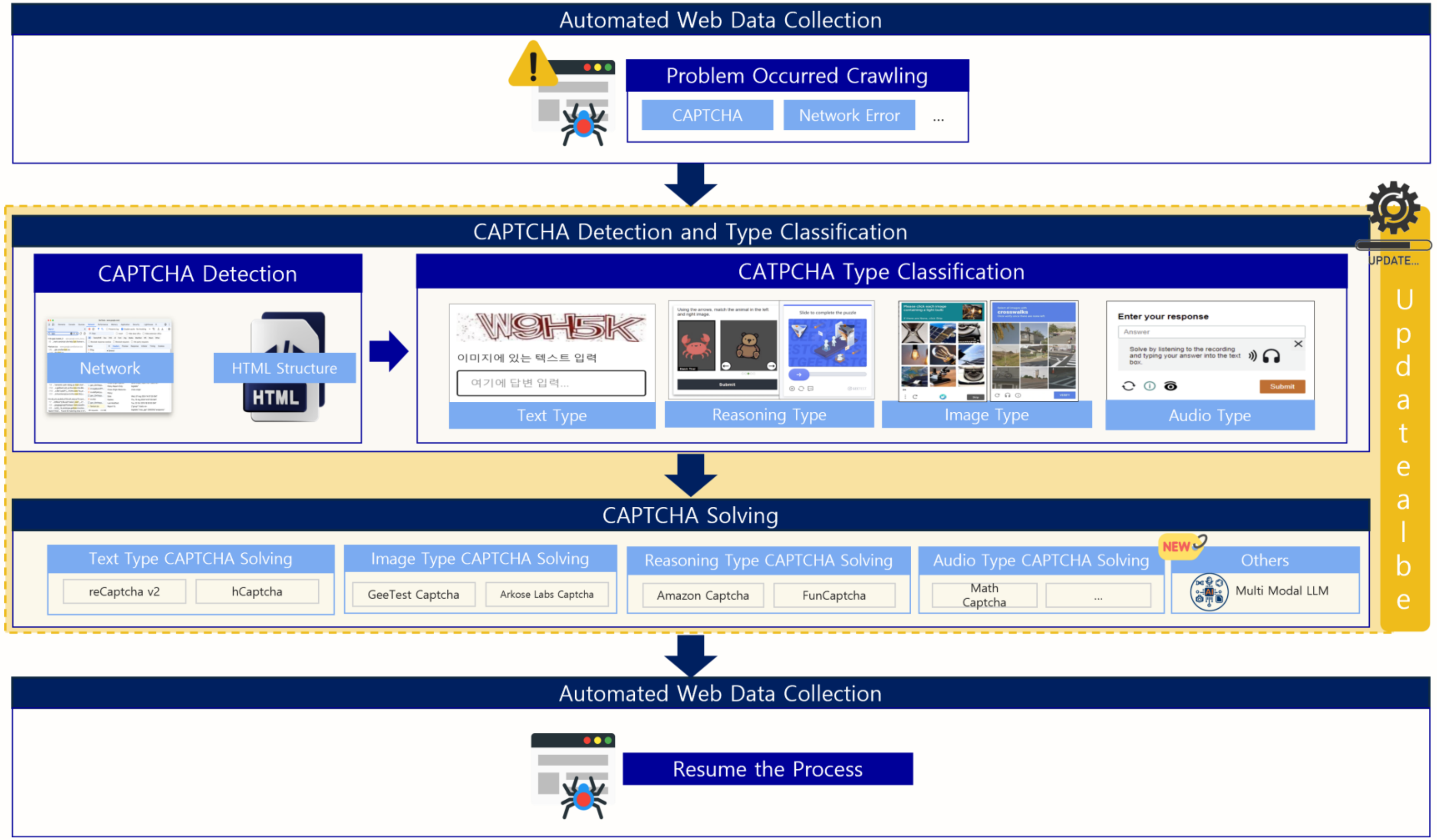
- Recent work uses LLMs and multimodal models to attack CAPTCHAs (e.g., decomposition + chain-of-thought approaches, and VLM-based methods), demonstrating partial success but also limitations in understanding and latency.
- Prior studies tend to focus on specific CAPTCHA types or propose generic models without detailed, end-to-end procedures for deployment.

Motivation & Objective

- CAPTCHAs hinder real-time, automated web-based evidence collection for digital forensics and incident response.
- Automated CAPTCHA bypassing can restore timeliness and scale to forensic data collection, improving crime prevention and investigation.
- Develop an open-source framework that detects and solves common CAPTCHA types by analyzing structural characteristics and applying ML/data-processing techniques.

2. Methodology

Methodology

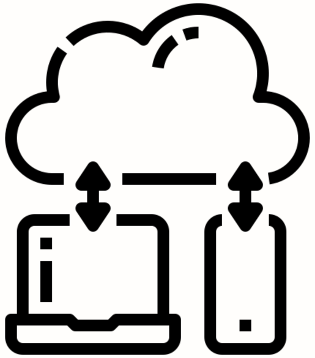

Automated Web Data Collection
Problem Occurred Crawling
CAPTCHA
Network Error
...
CAPTCHA Detection and Type Classification
CAPTCHA Detection
Network
HTML Structure
HTML
CATPCHA Type Classification
W9H5K
이미지에 있는 텍스트 입력
여기에 답변 입력...
Text Type
Using the arrows, match the animal in the left and right image.
Submit
Slide to complete the puzzle
Reasoning Type
crosswalks
Image Type
Enter your response
Answer
Solve by listening to the recording and typing your answer into the text box.
Submit
Audio Type
UPDATE...
Updateable
CAPTCHA Solving
Text Type CAPTCHA Solving
reCaptcha v2
hCaptcha
Image Type CAPTCHA Solving
GeeTest Captcha
Arkose Labs Captcha
Reasoning Type CAPTCHA Solving
Amazon Captcha
FunCaptcha
Audio Type CAPTCHA Solving
Math Captcha
...
NEW
Others
Multi Modal LLM
Automated Web Data Collection
Resume the Process

3. Detection Stage

Detection and Identification of CAPTCHA

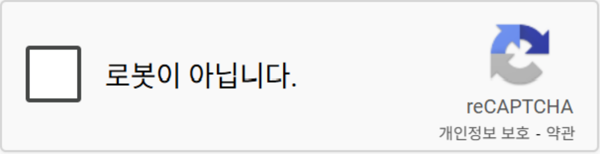
- Two CAPTCHA providers used in this research : reCAPTCHA v2, GeeTest
- Three CAPTCHA types tested : Image, Logical Reasoning (Slide, Icon)

- Differentiating each providers ➡ HTML Architecture
- Differentiating each types ➡ Network Record

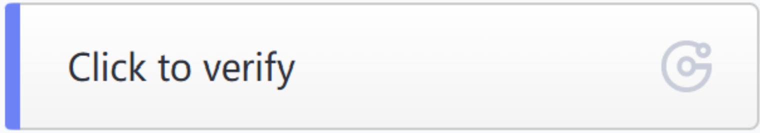


Detection of CAPTCHA

- For determining whether the cause of the interruption is due to CAPTCHA
- Utilize HTML Structure – Tags and Attributes
- reCAPTCHA v2
  - Search <iframe> tag with title attribute 'reCAPTCHA'
- GeeTest
  - Explore mutiple <div> tags including 'geetest' in class attribute



```
<iframe title="reCAPTCHA" src="https://www.google.com/recaptcha/api2/anchor">
```



```
<div class="geetest_captcha_ed6c8739 geetest_captcha geetest_float geetest_nextReady">
```

Identification of CAPTCHA

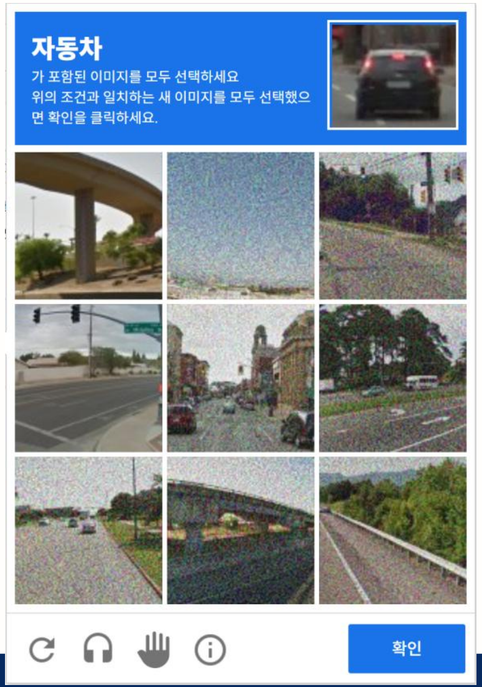
- Token information is used to send and receive CAPTCHA verification
- Network records are used to distinguish each CAPTCHA types

Types	Request URL
Slide	https://gcaptcha4.geetest.com/load?captcha_id=#&risk_type=slide
IconCrush	https://gcaptcha4.geetest.com/load?captcha_id=#&risk_type=match
Icon	https://gcaptcha4.geetest.com/load?captcha_id=#&risk_type=icon
Image	https://gcaptcha4.geetest.com/load?captcha_id=#&risk_type=nine

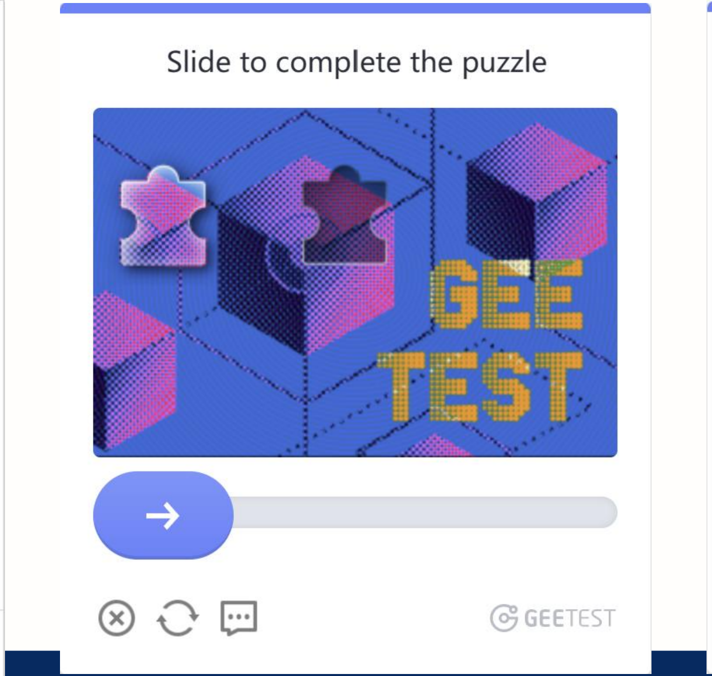
4. Resolution Stage

Resolution of each types of CAPTCHA

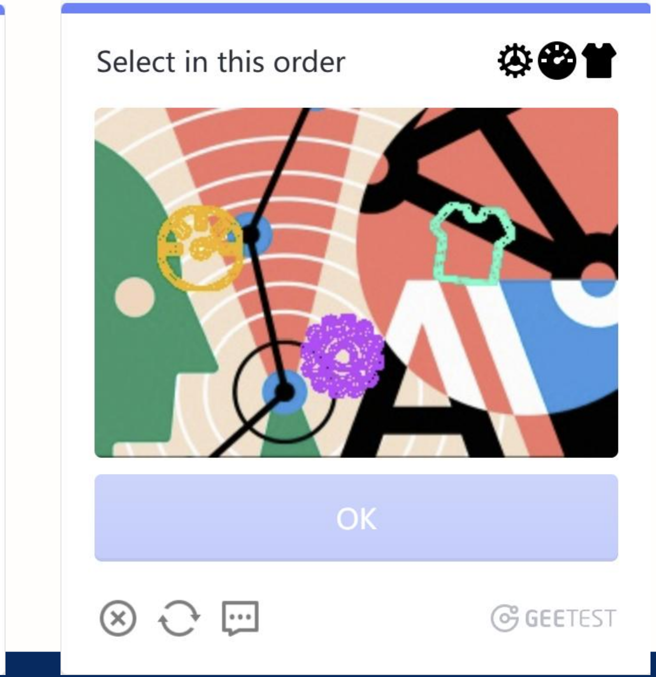
- This stage operates after determining the presence and type of CAPTCHA
- It covers the solution algorithm for each type,
  and the overall procedure of determining the returning the solution for image, slide, and icon type CAPTCHA



자동차
가 포함된 이미지를 모두 선택하세요
위의 조건과 일치하는 새 이미지를 모두 선택했으면 확인을 클릭하세요.



Slide to complete the puzzle

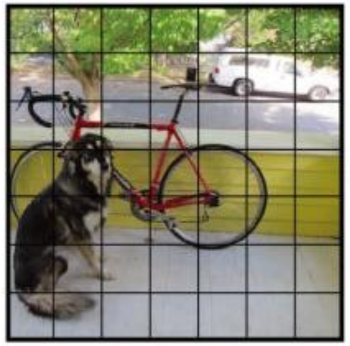


Select in this order

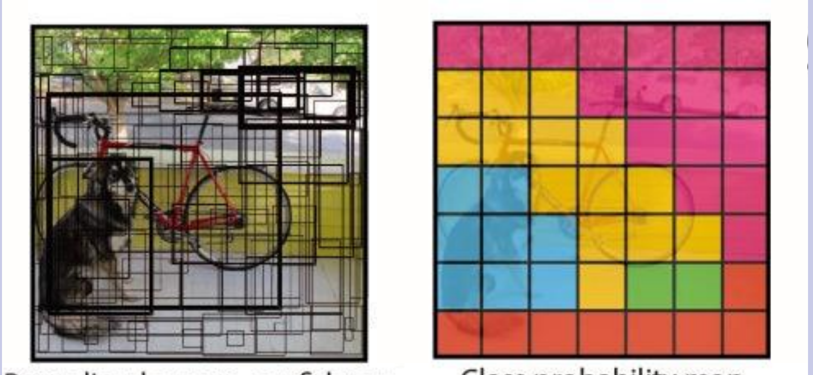
Resolution on Image type CAPTCHA

- Requires Object Classification & Bounding Boxes
  - YOLO (You Look Only Once) is the most suitable model !
- YOLO can classify multiple objects and locate simultaneously with a single operation

Fast Enough!



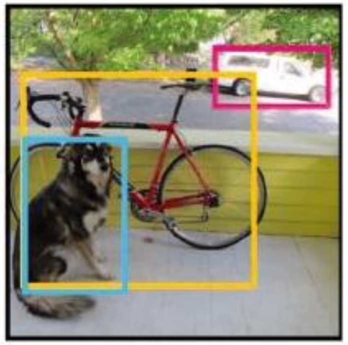
S × S grid on input



Bounding boxes + confidence

Class probability map

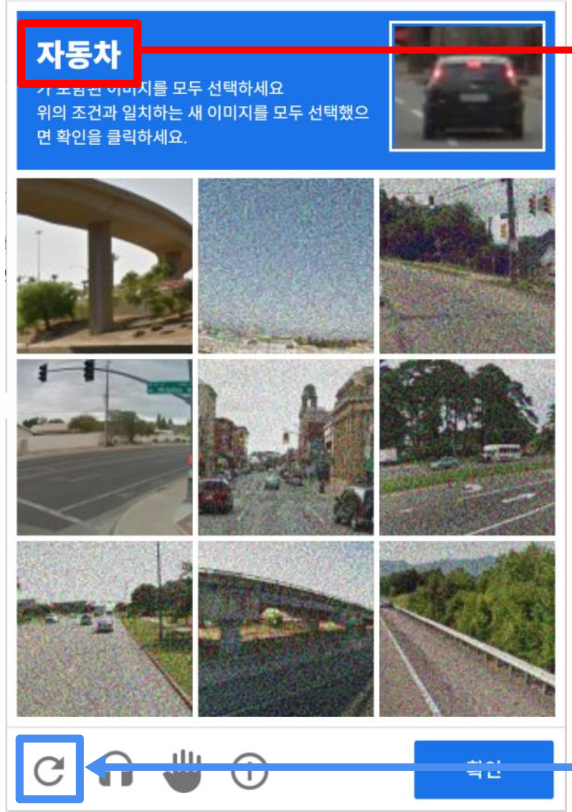
(x, y, w, h, class, confidence)



Final detections

Resolution on Image type CAPTCHA

- 1) Extract label from the Above Text



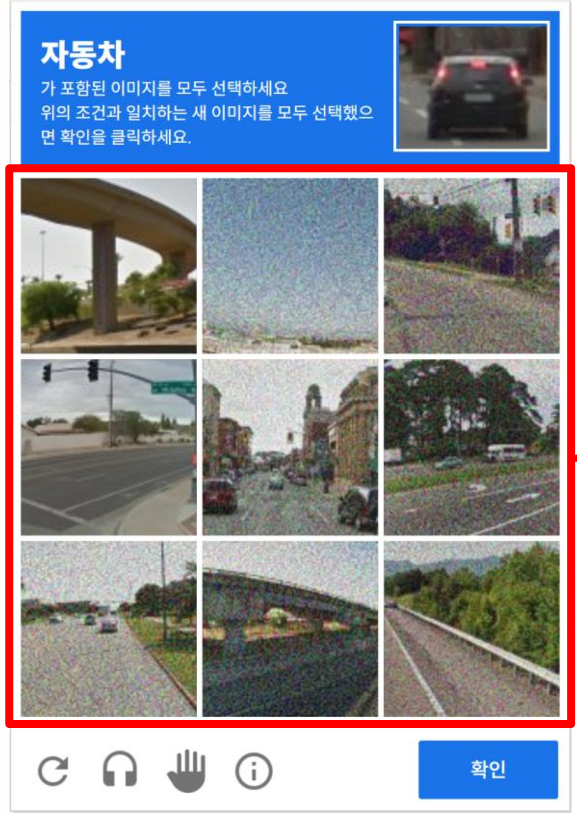
자동차
가 포함된 이미지를 모두 선택하세요
위의 조건과 일치하는 새 이미지를 모두 선택했으면 확인을 클릭하세요.

Has been already trained in the model?

If not, reload CAPTCHA

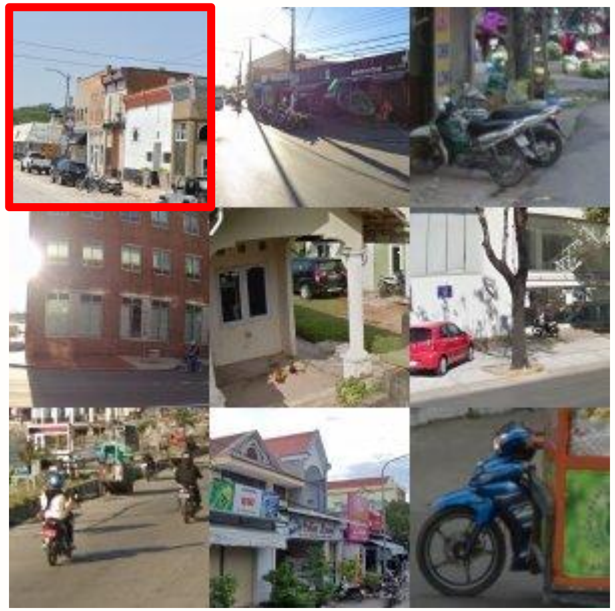
Resolution on Image type CAPTCHA

- 2) Download the main square image below



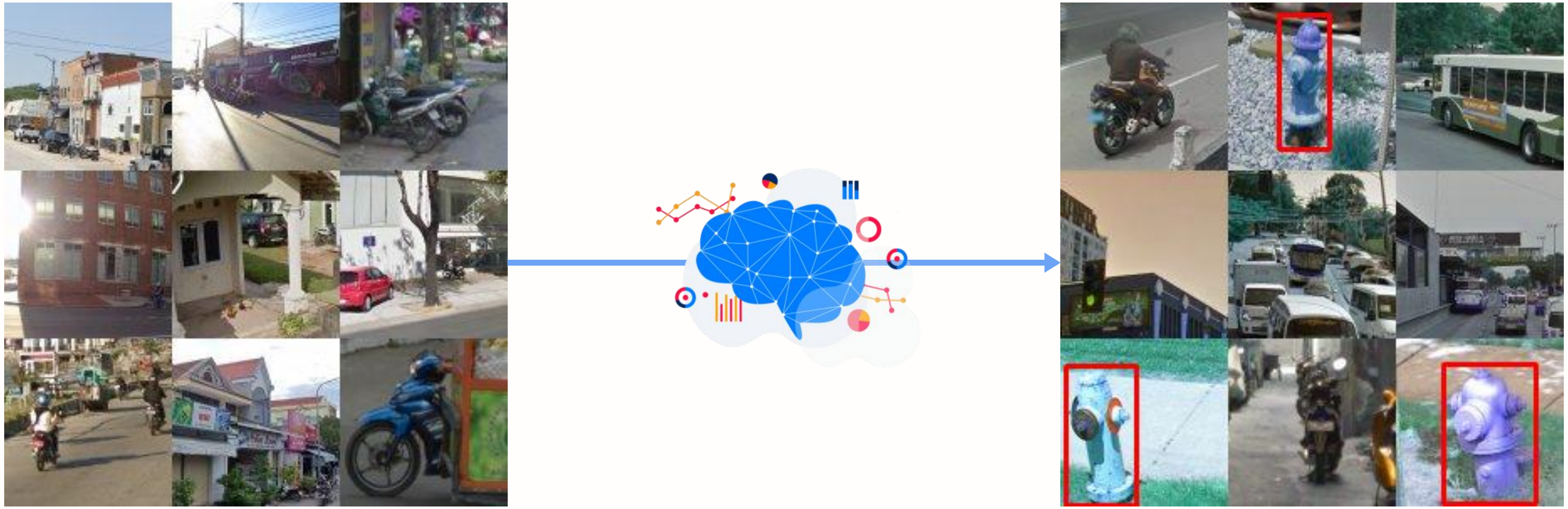
자동차
가 포함된 이미지를 모두 선택하세요
위의 조건과 일치하는 새 이미지를 모두 선택했으면 확인을 클릭하세요.

Each tiles has its own image URL



Resolution on Image type CAPTCHA

- 3) Feed the model with the image and target label



Resolution on Image type CAPTCHA

- 4) Select each tiles on a random sequence and time gap

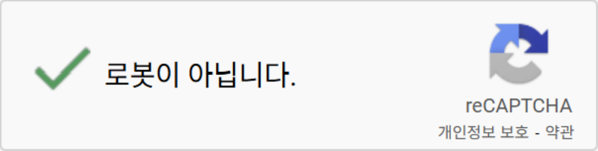
- 5) Check whether the CAPTCHA is solved
  - 1. Is there an iframe with the title attribute set to 'reCAPTCHA' ?
  - 2. Is the 'recaptcha-anchor' checkbox activated?

If at least one condition is satisfied,

Terminate the process and return 'True' to the program

If not,

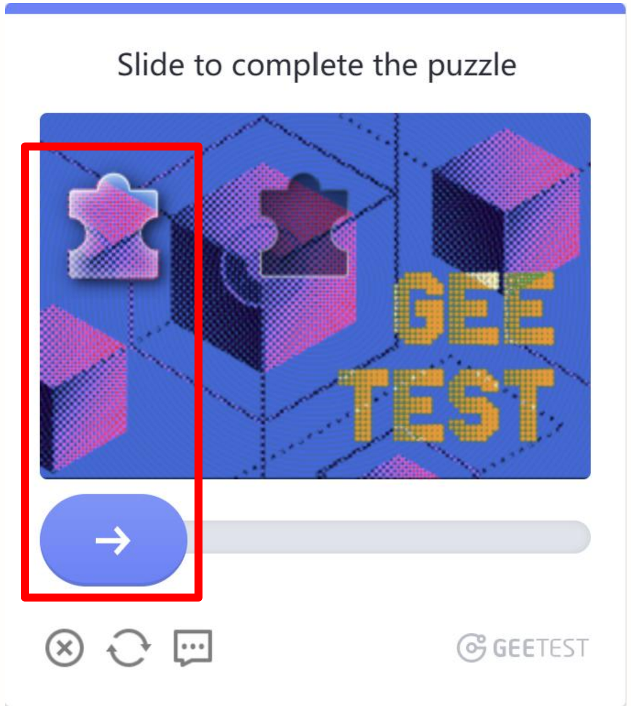
Return to the beginning and repeat the same process



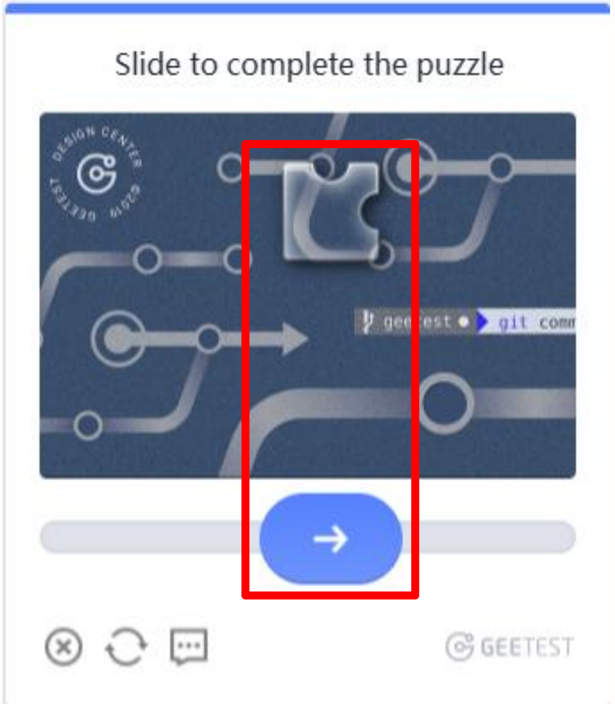
✓ 로봇이 아닙니다.

Resolution on Slide type CAPTCHA

- It requires users to ...
  - Find the correct position where the puzzle piece fits within the overall image
  - Drag the slider or the puzzle piece with the mouse to align it with that position



Slide to complete the puzzle

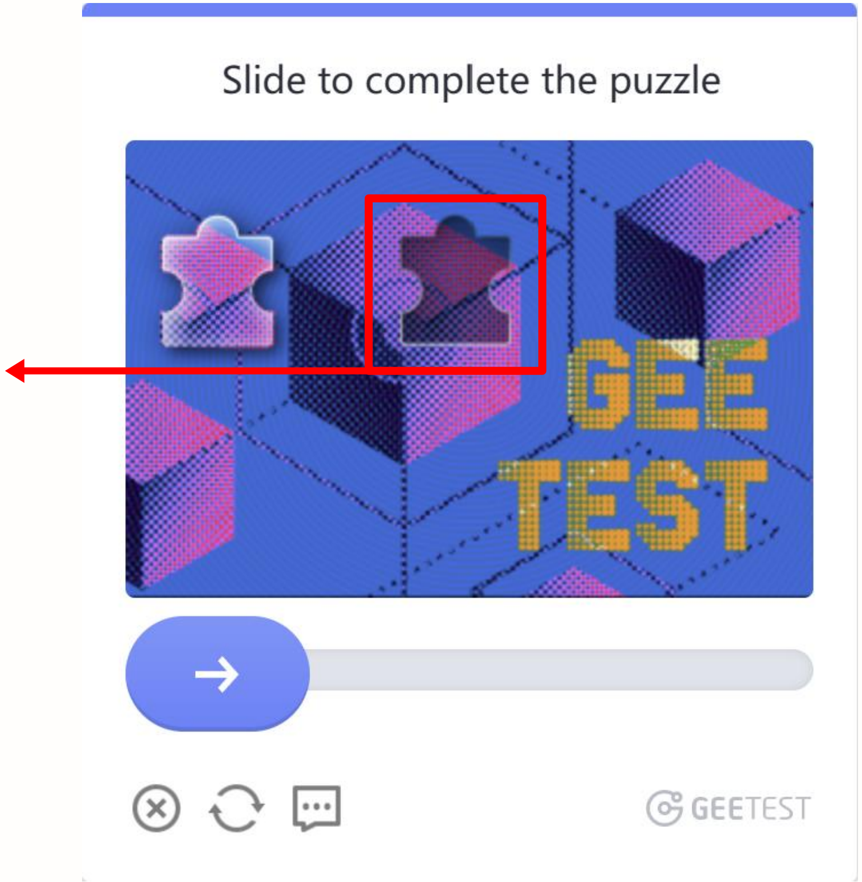


Slide to complete the puzzle

Resolution on Slide type CAPTCHA

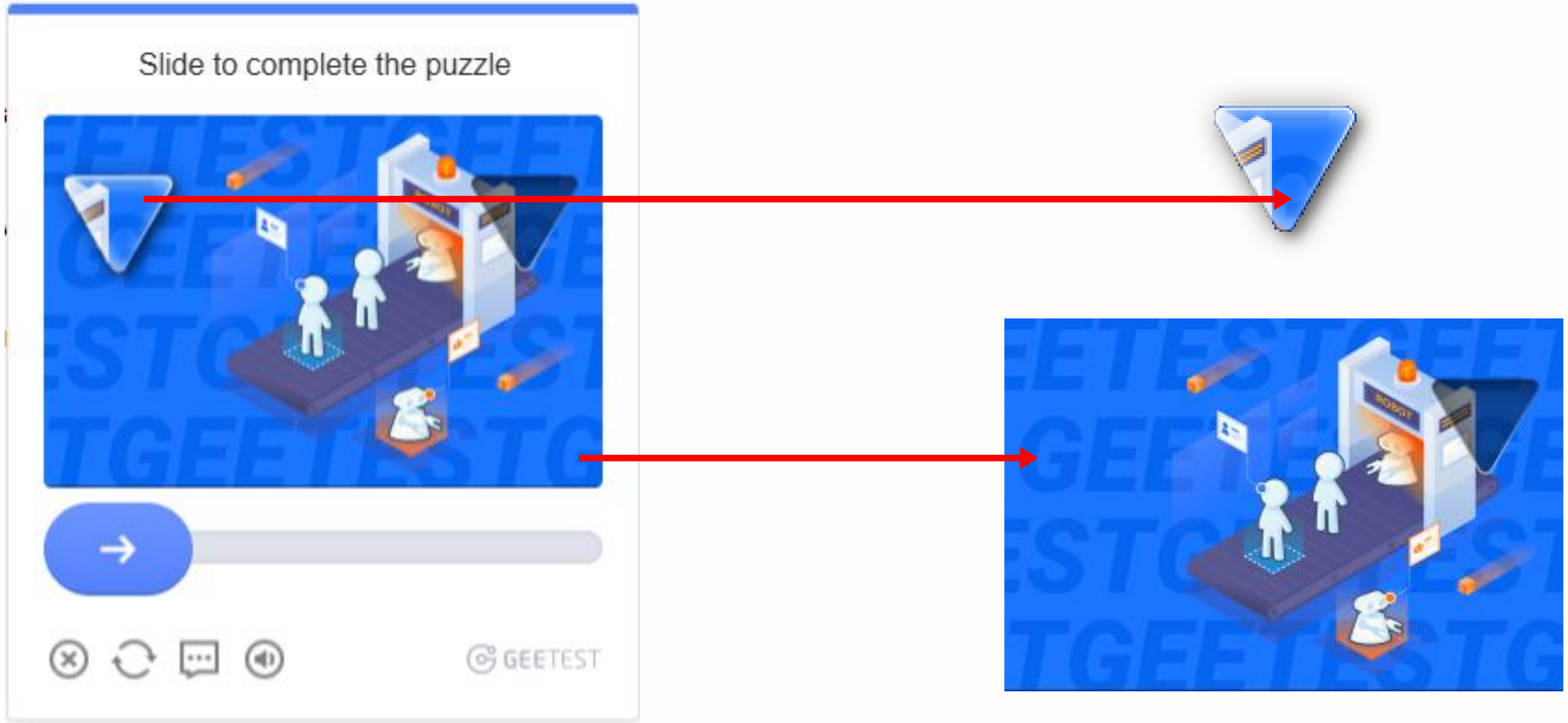
- A hint is provided in the image

The area where the puzzle piece fits is darker than the surrounding area


Slide to complete the puzzle

Resolution on Slide type CAPTCHA

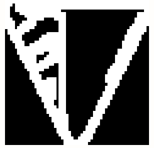
- 1) Download both the puzzle image and the background image separately

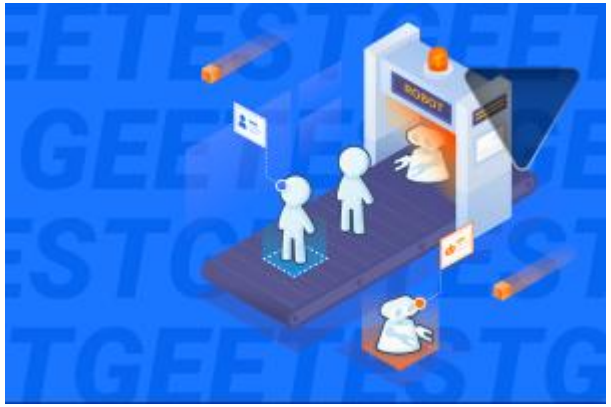


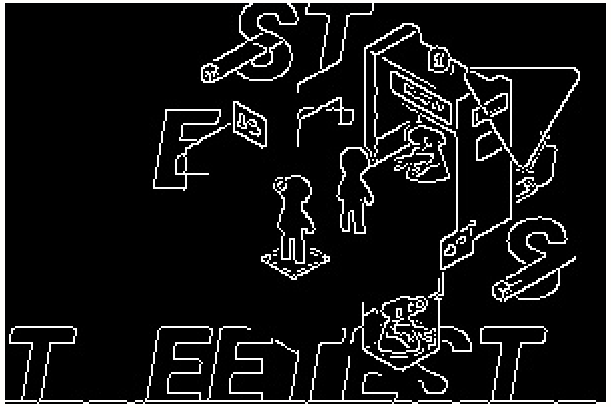
Slide to complete the puzzle

Resolution on Slide type CAPTCHA

- 2) Extract the contours of each image
  - To improve computational efficiency and accuracy

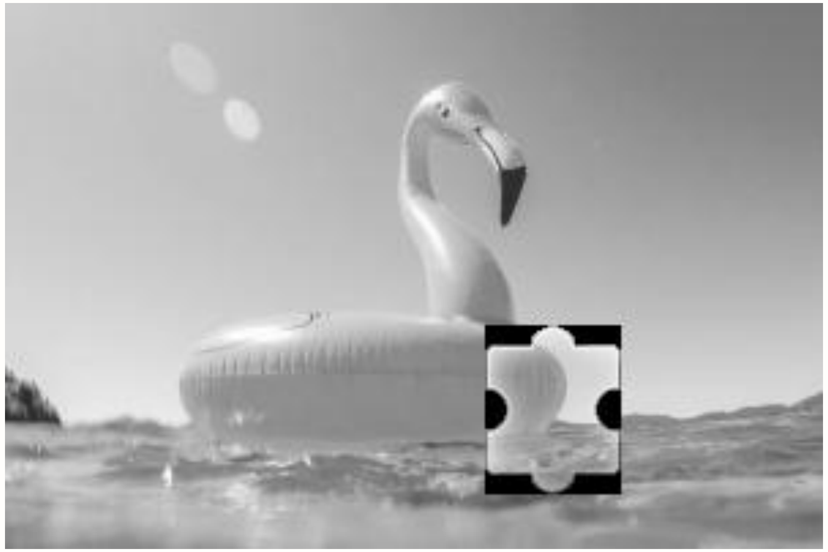






Resolution on Slide type CAPTCHA

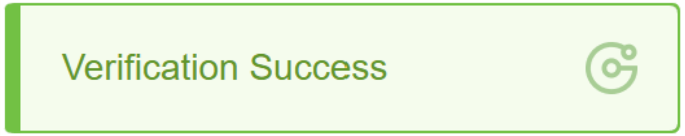
- 3) Perform template matching between two images,
  and calculate the distance to move the slider
  - Template Matching : Smaller image slides over the larger image to find regions that match the template
  - The similarity between the template and each region is measured, using correlation to find the best match



Resolution on Slide type CAPTCHA

- 4) Move the slider at random intervals

- 5) Check whether the CAPTCHA is solved
    - 1. Is the display property of div (geetest_box_wrap) set to none?
    - 2. Is the content of div (geetest_tip) “Verification Success”?



Verification Success

If at least one condition is satisfied,

Terminate the process and return ‘True’ to the program

If not,

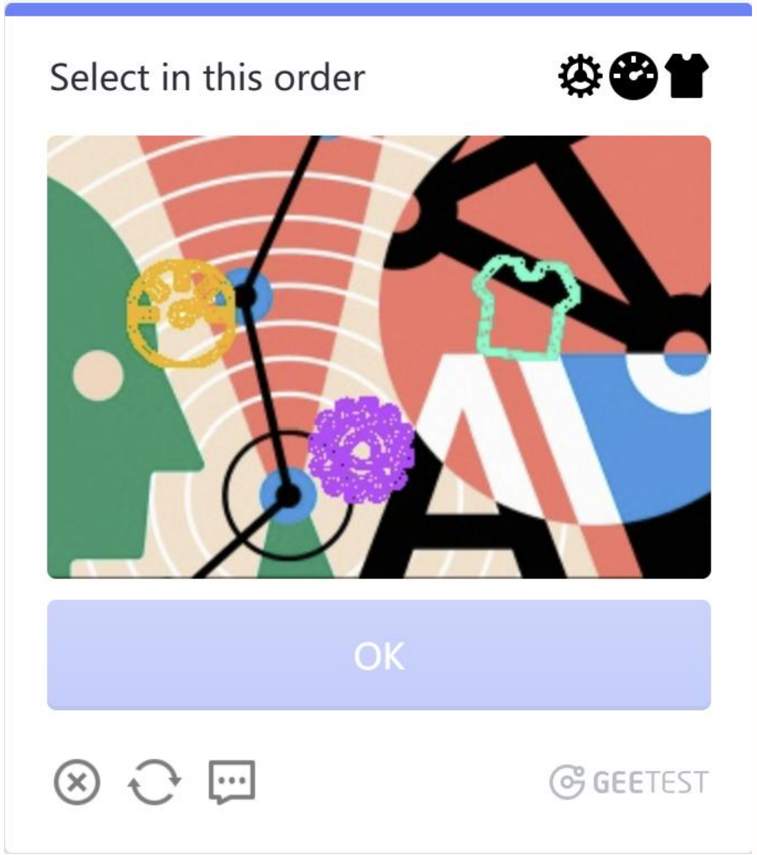
Return to the beginning and repeat the same process

Resolution on Icon type CAPTCHA

- It requires users to ...
  - Detect each icons from the below image
  - Select each icons in order as specified above

- How about utilizing Multi Modal LLM?
  - Limitations on performing multiple task simultaneously

    ex) Hard to recognize the concept sequence.
  - Difficult to detect precise coordinates



Select in this order

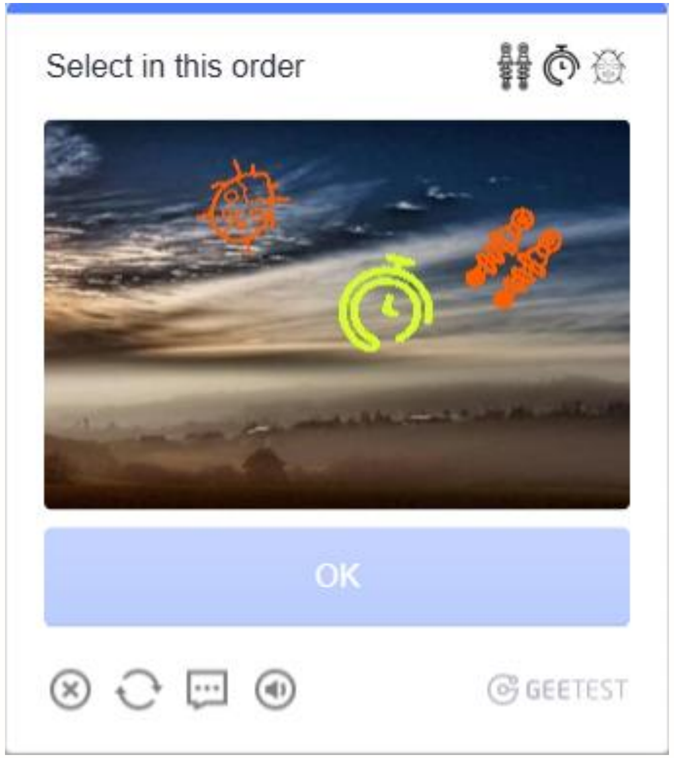
Resolution on Icon type CAPTCHA

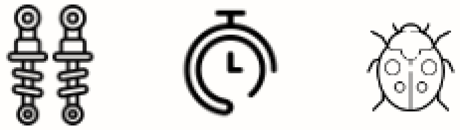
- Apply Chain Of Thought Prompting !

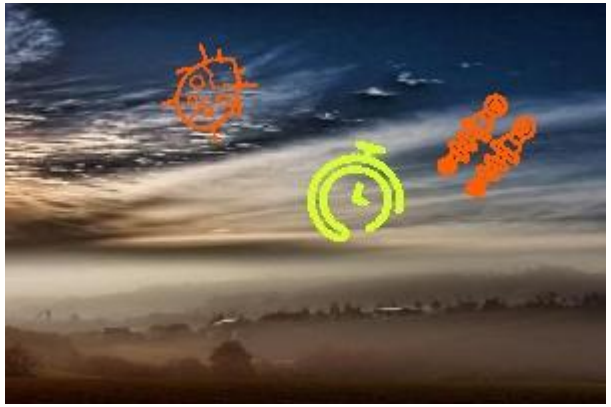
- MiniCPM-V-4_t-int4
  - A multimodal large language model developed by the OpenBMB team
  - Capable of understanding and reasoning over both text and images
  - Strong performance in object recognition and relational reasoning within images

Resolution on Icon type CAPTCHA

- 1) Download each of the given icon image and the background image separately

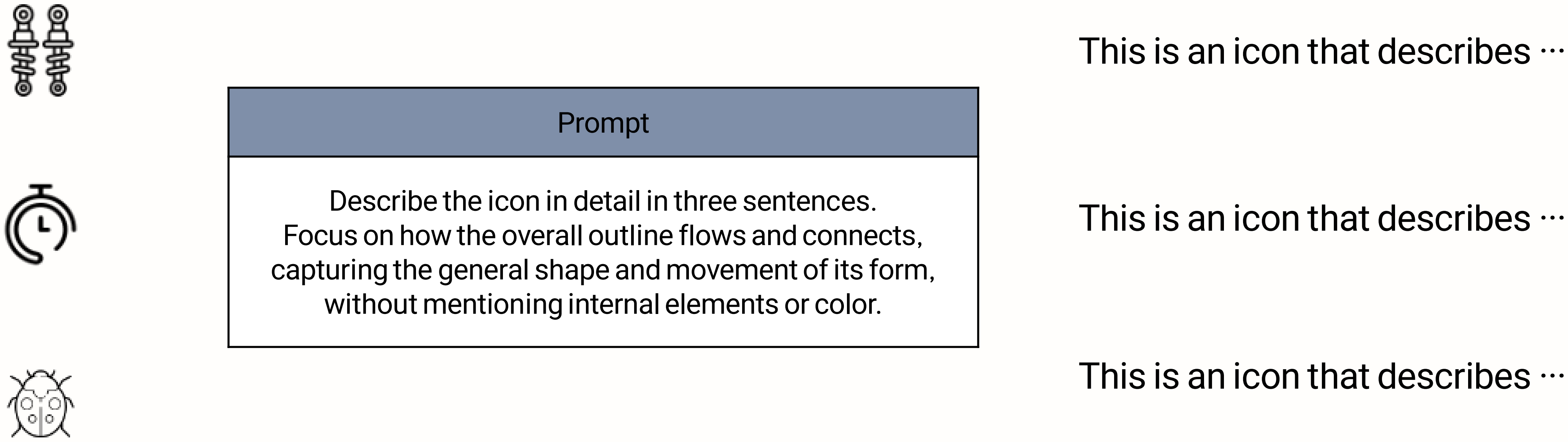






Resolution on Icon type CAPTCHA

- 2) Request LLM for detailed descriptions for each icon



Prompt

Describe the icon in detail in three sentences.
Focus on how the overall outline flows and connects,
capturing the general shape and movement of its form,
without mentioning internal elements or color.

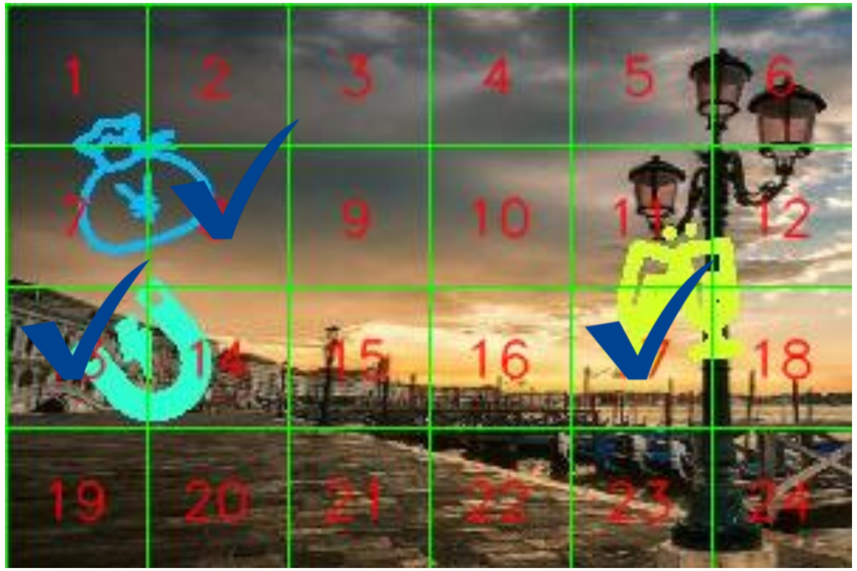
This is an icon that describes …

This is an icon that describes …

This is an icon that describes …

Resolution on Icon type CAPTCHA

- 3) Find the grid number containing the icon described in the text
  - The image is divided into 24 grids, to overcome the limitation that LLM can't guess exact coordinates

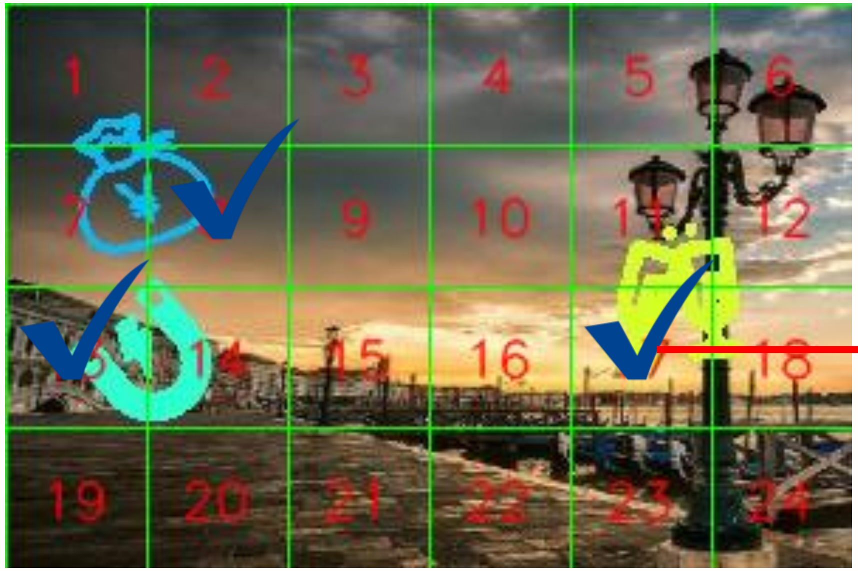


This is an icon that describes …

Prompt
This is an explanation of a specific icon : {icon}. Find it and return the number of the grid that contains that icon. If multiple grids contain the icon, return only one that contains the center of the icon. Ensure that the result is only the number, with no extra text or formatting.

Resolution on Icon type CAPTCHA

- 3) Find the grid number containing the icon described in the text

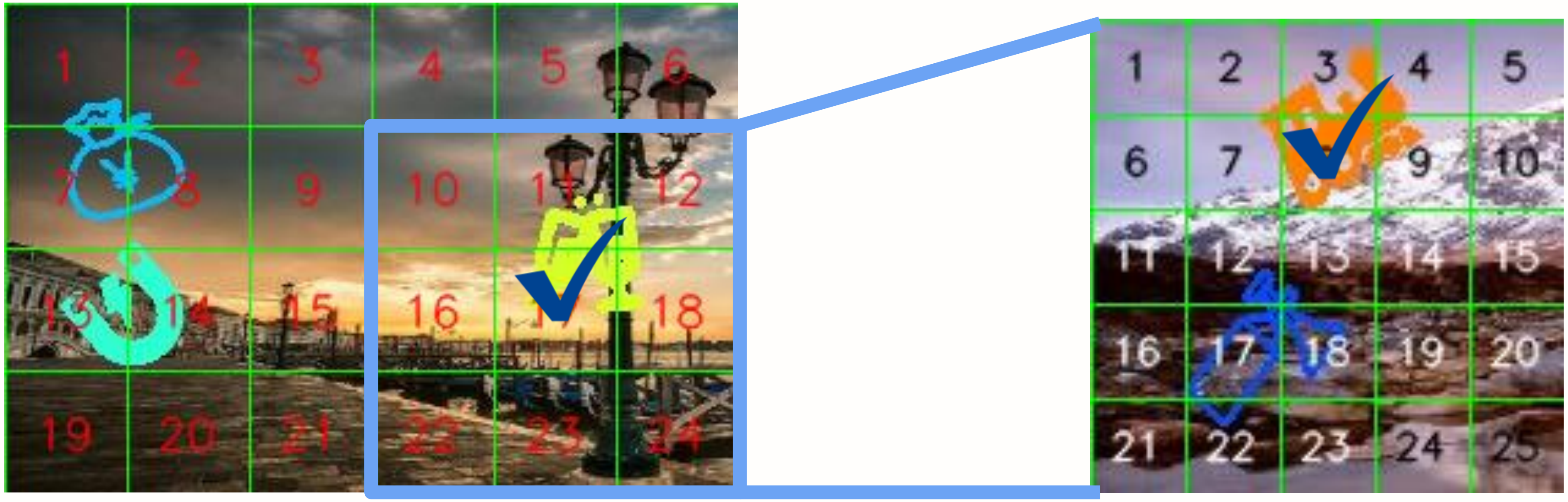


This isn’t enough!

There are cases
where the icons spans multiple grids
or the model returns the number of nearby grid instead

Resolution on Icon type CAPTCHA

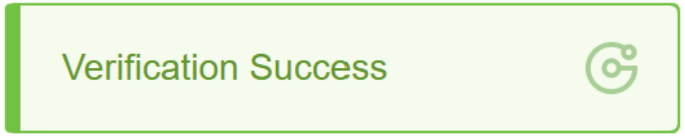
- 4) Extract a more detailed image that includes
  the selected grid and it surrounding area,
  divide it into 25 grids and repeat the same process



Resolution on Icon type CAPTCHA

- 4) Click each icon in order as specified

- 5) Check whether the CAPTCHA is solved
  - 1. Is the display property of div (geetest_box_wrap) set to none ?
  - 2. Is the content of div (geetest_tip) "Verification Success"?



Verification Success

If at least one condition is satisfied,

Terminate the process and return 'True' to the program

If not,

Return to the beginning and repeat the same process

5. Evaluation

Evaluation Environments

Environment (1)	CPU	Snapdragon X1E-80-100
	RAM	16GB
	GPU	Qualcomm Adreno X1-85
	OS	Windows 11 Home
Environment (2)	CPU	Intel i9-10980XE
	RAM	256GB
	GPU	NVIDIA RTX 3090 24GB * 4
	OS	Ubuntu 20.04 LTS

Evaluation Indicators and Methods

- Success Rate
  - Judge whether the framework can accurately detect and solve different CAPTCHA types across 20 repeated trials.
- Inference Time
  - Refers to the average computation time required for each CAPTCHA type ex) from model prediction or template matching to coordinate calculation.
- Total Process Time
  - Represents the overall duration from CAPTCHA detection to the completion of the solving process.

Evaluation Result

Environments	Indicators	Image	Slide	Icon
Environment (1)	Success Rate	100% (20 / 20)	85% (17 / 20)	-
	Inference Time	22.5 s	0.06 s	-
	Process Time	45.2 s	22.9 s	-
Environment (2)	Success Rate	100% (20 / 20)	90% (18 / 20)	45% (9 / 20)
	Inference Time	0.56 s	0.02 s	20.8 s
	Process Time	19.7 s	19.9 s	42.4 s

6. Conclusion

Limitations

- The framework showed reduced performance in environments with limited hardware or slow computation, where LLM-based models could not be applied effectively

  ➡ To address this, a dedicated GPU server is proposed, where computations are offloaded upon CATPCHA detection.

- Although LLM with Chain-of-Thought reasoning and grid-based search improved icon-type CAPTCHA solving, textual icon descriptions alone were insufficient for precise localization.

  ➡ Future research will explore multi-image models and LLM fine-tuning with rotated and distorted image pairs to improve spatial understanding.

Contributions

- Proposed an integrated framework for CAPTCHA detection and solving, enabling automated web data collection and analysis without human intervention.
- Provided detailed detection and solving procedures verified through experiments, offering a practical guideline for future CAPTCHA types.
- Demonstrated that even non-specialized open-source LLMs can perform recognition and reasoning tasks confirming their potential in security and authentication contexts.

Future Work

- The framework will be continuously expanded to support a wider range of CAPTCHA types beyond image, slide, and icon based formats.
- Future studies will explore the integration of large language models (LLMs) to develop a unified approach applicable to most CAPTCHA types.
- Additionally, comparative analyses will be conducted between type-specific algorithms and LLM-based methods to evaluate performance and efficiency.

Thank You

Presented by：

JunBeom Cho

Cyber Security & Digital Forensics Lab.