

Автоматизация игры с использованием компьютерного зрения и распознавания текста

Андрюенко Иван Сергеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье описана автоматизация игры "1+2=3" с использованием компьютерного зрения и распознавания текста. В ходе исследования была разработана программа на языке программирования python, способная захватывать скриншоты области экрана с графическими надписями игры и распознавать текст на изображении. Для распознавания текста был использован модуль pytesseract, основанный на алгоритмах оптического распознавания символов. Полученный текст подвергся обработке и вычислению. В случае совпадения результата с predefined ответами игры, программа автоматически кликала на соответствующую кнопку на экране.

Ключевые слова: компьютерное зрение, автоматизация игры, распознавание текста, Python, pyautogui, pytesseract.

Game automation using computer vision and text recognition

Andrienko Ivan Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article describes the automation of the game "1 +2 =3" using computer vision and text recognition. During the research, a program was developed in the python programming language, capable of capturing screenshots of an area of the screen with graphic inscriptions of the game and recognizing text on the image. The pytesseract module based on optical character recognition algorithms was used for text recognition. The resulting text was processed and calculated. If the result coincided with the predefined answers of the game, the program automatically clicked on the corresponding button on the screen.

Keywords: computer vision, game automation, text recognition, Python, pyautogui, pytesseract.

1 Введение

1.1 Актуальность

В современном мире компьютерное зрение и распознавание текста играют важную роль во многих областях, включая игровую индустрию.

Автоматизация игр с помощью этих технологий представляет собой актуальную и интересную тему исследования. Автоматизация игры "1+2=3" в телеграмме с использованием компьютерного зрения и распознавания текста имеет практическую значимость для игроков, которые стремятся повысить свою эффективность и достичь лучших результатов в игре. Такой подход позволяет сократить время, затрачиваемое на решение математических уравнений и нажатие на правильные ответы, что особенно важно в ситуациях, когда игра ограничена по времени. Кроме того, данная автоматизация игры может быть применена и в других контекстах, где требуется распознавание текста и выполнение соответствующих действий. Это может включать автоматизацию других игр с похожими механиками, а также применение в других сферах.

1.2 Обзор исследований

Р.А. Кравец, К.С. Бараш и М.Д. Карачун исследовали использование языка Python для автоматизации процесса получения моделей прогнозирования надежности выборок полупроводниковых приборов. В работе описывается применение языка Python для сбора и обработки данных в области полупроводниковых приборов [1]. Е.А. Комзалов исследовал автоматизацию процесса предприятия с использованием языка программирования Python. Описано применение языка Python для автоматизации различных процессов предприятия [2]. А.Д. Суховаров представил методику обнаружения и извлечения текста из изображения с использованием библиотеки PyTesseract. Представлены подробности использования библиотеки PyTesseract для распознавания текста на изображениях [3]. Е.В. Губкина, К.Е. Шишкин и К.Г. Янковская исследовали автоматизацию некоторых процессов работы с текстовыми документами. В работе представлены примеры использования языка Python и различных библиотек для автоматизации операций с текстовыми документами [4]. А.М. Бабаев, Ю.В. Алексеенко, А.М. Бабаев и Т.Г. Авдеева исследовали автоматизацию извлечения текста из изображений с помощью оптического распознавания символов. Описываются методы и инструменты для извлечения текста из изображений с использованием оптического распознавания символов [5].

1.3 Цель исследования

Целью данного исследования является автоматизация игры "1+2=3" в телеграмме с помощью компьютерного зрения и распознавания текста.

2 Материалы и методы

Для реализации автоматизации игры была использована среда программирования PyCharm Community Edition 2022.1.3. Языком программирования был выбран Python, так как он обладает мощными библиотеками для обработки изображений и распознавания текста. В процессе разработки программного решения, был использован модуль pyautogui для

захвата скриншотов области экрана. Для распознавания текста на изображении была применена библиотека pytesseract.

3 Результаты и обсуждение

В игре "1+2=3" игроку предлагаются математические уравнения, в которых необходимо правильно решить простые арифметические операции. Игра использует сложение и вычитание. Цель игры заключается в том, чтобы максимально быстро и точно решить уравнение и выбрать правильный ответ из трех вариантов. Игра предоставляет ограниченное время на решение каждого уравнения, и игрок должен нажать на правильный ответ до истечения времени. Успешное решение уравнений позволяет набрать очки и продвигаться в игре, а неправильные ответы или время, истекшее без решения, приводят к поражению.

В программе планируется создать автоматизированную систему для игры "1+2=3". Программа будет основываться на использовании компьютерного зрения и распознавания текста для решения математических уравнений на экране игры. После захвата скриншота области экрана с графическими надписями, программа будет распознавать текст на изображении с помощью библиотеки pytesseract. Затем произойдет вычисление результатов для уравнений и сравнение с ожидаемыми ответами. При правильном решении игра будет автоматически продолжаться с нажатием на соответствующую кнопку на экране. Код программы будет запускаться в бесконечном цикле и будет ждать нажатия клавиши "Пробел" для запуска процесса игры и распознавания текста.

Импортируем необходимые библиотеки: `import pyautogui, import pytesseract, import keyboard, import re`. Библиотека «pyautogui» используется для захвата скриншотов и взаимодействия с экраном. Библиотека «pytesseract» используется для распознавания текста на изображении. Библиотека «keyboard» используется для регистрации нажатия клавиши. Библиотека «re» используется для обработки строк. Эти библиотеки являются необходимыми для выполнения дальнейших операций, связанных с автоматизацией игры и распознаванием текста.

Далее необходимо определить область экрана, которая будет содержать уравнение, с помощью переменной «region1». Эта переменная представляет собой кортеж, содержащий координаты и размеры прямоугольной области на экране.

Перед использованием модуля распознавания текста необходимо установить и настроить Tesseract OCR. Tesseract OCR является библиотекой для распознавания оптического текста и используется для извлечения текста из изображений. В данном коде для статьи была выполнена предварительная установка Tesseract OCR на компьютер, а затем был указан путь к исполняемому файлу в переменной. Это позволяет модулю «pytesseract» обращаться к функциональности Tesseract OCR для выполнения распознавания текста на изображениях. Затем инициализируется модуль распознавания текста «pytesseract», устанавливая путь к исполняемому файлу

распознавания текста Tesseract OCR. Эти действия позволяют программе определить и охватить нужные области на экране для последующего распознавания текста (рис.1).

```
1 import pyautogui
2 import pytesseract
3 import keyboard
4 import re
5 |
6 region1 = (710, 560, 500, 50)
7
8 pytesseract.pytesseract.tesseract_cmd = r'C:/Program Files/Tesseract-OCR/tesseract.exe'
9
```

Рисунок 1 – Создание необходимых переменных

Определяем функцию «perform_text_recognition()», которая выполняет процесс распознавания текста на изображении с графическим текстом. Сначала функция захватывает скриншот определенной области экрана, которая была определена ранее, с помощью функции «pyautogui.screenshot()». Затем изображение передается в функцию «pytesseract.image_to_string()», которая осуществляет распознавание текста с использованием Tesseract OCR. Результат распознавания сохраняется в переменной «text1». Далее применяется регулярное выражение с помощью модуля «re», чтобы удалить все символы, кроме цифр 1, 2, 3 и знаков плюса и минуса. Текст выводится на экран с помощью функции «print()». Этот код представляет ключевую часть автоматизированного процесса распознавания текста в игре (рис. 2).

```
10
11 def perform_text_recognition():
12     # Захват скриншота области экрана с графическими надписями
13     image = pyautogui.screenshot(region=(region1[0], region1[1], region1[2], region1[3]))
14
15     # Распознавание текста на изображении
16     text1 = pytesseract.image_to_string(image, config='--psm 6 --oem 3 -l eng')
17
18     text1 = re.sub(r"^[123+-]", "", text1)
19     print(text1)
```

Рисунок 2 – Создание автоматизированного процесса

Далее выполняется вычисление результатов для распознанных выражений, которые были получены на предыдущем шаге. Результат распознавания текста, сохраненный в переменной «text1», используется с помощью функции «eval()» для вычисления математического выражения. Затем происходит сравнение полученного результата «result1» с числами 1, 2 и 3. В зависимости от результата сравнения, используя библиотеку «pyautogui», производится клик мыши в определенной позиции на экране. В данном случае, если результат равен 1, будет выполнен клик в позиции (960, 650), там находится кнопка ответа, если результат равен 2 - в позиции (960, 720), а если результат равен 3 - в позиции (960, 800). Этот код отражает процесс

определения результата выражений и выполнения соответствующих действий в игре (рис. 3).

```
20
21     # Вычисление результатов для выражений
22     result1 = eval(text1)
23
24     # Определение результата сравнения текстов
25     if result1 == 1:
26         pyautogui.click(960, 650)
27     elif result1 == 2:
28         pyautogui.click(960, 720)
29     elif result1 == 3:
30         pyautogui.click(960, 800)
```

Рисунок 3 – Вычисление и выбор правильного ответа

Далее реализуем бесконечный цикл, который ожидает нажатия клавиши "пробел" с помощью функции «`keyboard.wait()`». Как только пользователь нажимает клавишу "пробел", вызывается функция «`perform_text_recognition()`», которая выполняет процесс распознавания текста на экране и принимает соответствующие действия в игре на основе распознанного результата. Этот цикл позволяет программе ожидать и реагировать на каждое нажатие клавиши "пробел", обеспечивая автоматизацию игрового процесса на основе распознавания текста (рис. 4).

```
32
33     while True:
34         keyboard.wait('space')
35         perform_text_recognition()
36
```

Рисунок 4 – Реализация бесконечного цикла

Проверяем разработанную программу на игре. Программа практически моментально отвечает на уравнение правильным ответом «1» (рис. 5). Также в терминал программа выводит уравнения, которые распознает во время игры (рис. 6).

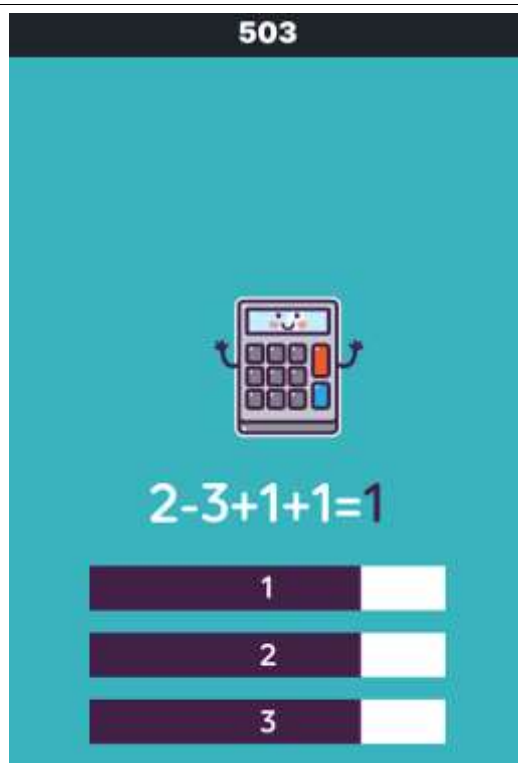


Рисунок 5 – Правильный ответ программы

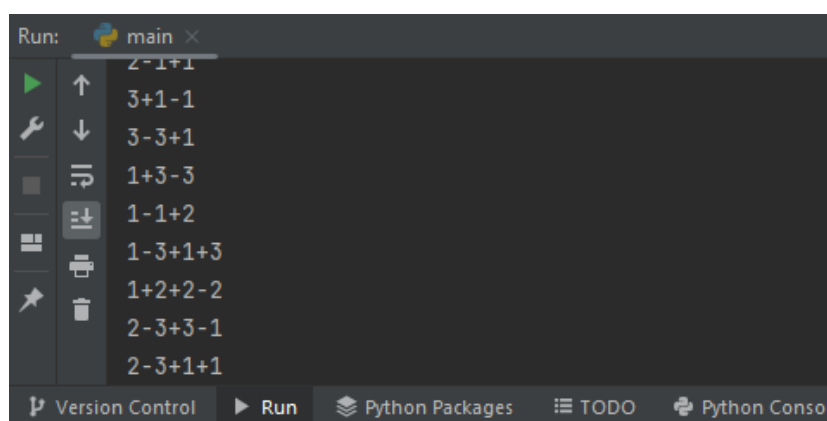


Рисунок 6 – Вывод игровых задач в консоль

Выводы

В данной научной статье исследована и разработана автоматизированная система для игры с использованием компьютерного зрения и распознавания текста. Был представлен код программы, который осуществляет захват и обработку скриншотов с графическими надписями, распознавание текста на изображениях с помощью библиотеки pytesseract, а также вычисление и сравнение результатов выражений. Программа реагирует на нажатие клавиши "пробел" и автоматически выполняет соответствующие действия в игре. Результаты исследования подтверждают эффективность использования компьютерного зрения и распознавания текста в автоматизации игровых процессов. Предложенный подход может быть

применен в других играх или задачах, требующих анализа и обработки текстовых данных для автоматизации действий.

Библиографический список

1. Кравец Р.А., Бараш К.С., Карачун М.Д. Использование языка Python для автоматизации процесса получения моделей прогнозирования надёжности выборок полупроводниковых приборов // Электронные системы и технологии. Сборник материалов 58-й научной конференции аспирантов, магистрантов и студентов БГУИР. Минск, 2022. С. 203-207.
2. Комзалов Е.А. Автоматизация процесса предприятия с использованием языка программирования Python // Внедрение передового опыта и практическое применение результатов инновационных исследований. Сборник статей Международной научно-практической конференции. Уфа, 2021. С. 68-71.
3. Суховаров А.Д. Методика обнаружения и извлечения текста из изображения с использованием библиотеки PyTesseract // Молодость. Интеллект. Инициатива. Материалы XI Международной научно-практической конференции студентов и магистрантов. Витебск, 2023. С. 70-71.
4. Губкина Е.В., Шишкин К.Е., Янковская К.Г. Автоматизация некоторых процессов по работе с текстовыми документами // Информация и образование: границы коммуникаций. 2021. № 13 (21). С. 123-127.
5. Бабаев А.М., Алексеенко Ю.В., Бабаев А.М., Авдеева Т.Г. Автоматизация извлечения текста из изображения посредством оптического распознавания символов // Особенности современного этапа развития естественных и технических наук. Сборник научных трудов по материалам Международной научно-практической конференции. В 2-х частях. Под общей редакцией Е.П. Ткачевой. 2018. С. 31-36.
6. Пишем бот для пазл игры на Python/Хабр: <https://habr.com/ru/articles/525380/> (дата обращения: 28.06.2023)