

Інформаційні технології

УДК 000.04

Піпич Артем Андрійович

бакалавр комп'ютерних наук,

Національний технічний університет України

«Київський політехнічний інститут»

Пипич Артём Андреевич

бакалавр компьютерных наук,

Национальный технический университет Украины

«Киевский политехнический институт»

Pipich A.

Bachelor of computer science,

The National Technical University of Ukraine

«Kyiv Polytechnic Institute»

**ПОРІВНЯННЯ ЗАСОБІВ БАЗОВАНИХ НА ПОШУКУ ГРАФІЧНИХ
ЕЛЕМЕНТІВ ДЛЯ РОЗПІЗНАВАННЯ ЕЛЕМЕНТІВ ГРАФІЧНОГО
ІНТЕРФЕЙСУ КОРИСТУВАЧА**

**СРАВНЕНИЕ СРЕДСТВ ОСНОВАННЫХ НА ПОИСКЕ
ГРАФИЧЕСКИХ ЭЛЕМЕНТОВ ДЛЯ РАСПОЗНАВАНИЯ
ЭЛЕМЕНТОВ ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ПОЛЬЗОВАТЕЛЯ
COMPARISON SEARCH TOOLS BASED ON GRAPHIC ELEMENTS TO
IDENTIFY ELEMENTS OF A GRAPHICAL USER INTERFACE**

Анотація: розглянуто засоби розпізнавання елементів графічного інтерфейсу, проведений їх порівняльний аналіз, та визначено найкращий засіб, в розглянутому контексті, для реалізації відповідного програмного засобу.

Ключові слова: комп'ютерний зір, графічний інтерфейс користувача, ГКІ розпізнавання, автоматизація тестування, розробка аналізатора.

Аннотация: рассмотрены средства распознавания элементов графического интерфейса, произведен их сравнительный анализ, и определено лучшее средство для реализации программного модуля их устранения.

Ключевые слова: компьютерное зрение, пользовательский графический интерфейс, распознавание ГПИ, автоматизация тестирования, разработка анализатора.

Abstract: The means of identification GUI spent their comparative analysis, and determined the best way, in the above context, to implement the appropriate software.

Keywords: computer vision, graphical user interface, GUI recognition, test automation, analiser development.

Серед найпотужніших засобів автоматизації тестування на основі пошуку графічних елементів слід виділити Ranorex, SikuliX та T-Plan.

Ranorex став досить розповсюдженим завдяки надійності роботи на Windows системах. Будь-який крок також може бути написаний, або виправлений вручну [1]. Даний продукт має власну IDE: Ranorex Studio. В результаті організація коду і тест-кейсів максимально нагадує MS Visual Studio: усі тест-кейси розкладені по проектах; проекти, які використовуються разом, можна об'єднувати в рішення. На виході після компіляції ми отримуємо по одному .exe файлу на рішення і по одному .dll на проект [2]. Досить скопіювати їх на тестову машину і запустити як звичайний Windows додаток подвійним кліком миші.

Ranorex зберігає увесь код, в тому числі створений авто-генерацією, і усі репозиторії в текстовому виді (наприклад для репозиторіїв використовується формат XML) це дає можливість для версійності і спільної розробки використовуючи VCS, як наприклад Git, з усіма його можливостями. Ціна однієї ліцензії з прив'язкою до робочого місця коштує

1 480 € за купівлю і додатково ще 290 € за кожен рік користування (починаючи з другого) за подальші оновлення і підтримку.

SikuliX написано на мові програмування Java. Даний програмний продукт не потребує окремої машини для можливості проведення тестування, тобто може тестувати програму на тій же машині, де він запущений [8]. Єдині вимоги до машини - це наявність екрану і встановлена JRE 5 версії і новіше.

Обов'язковим для засобу автоматизації даного рівня є наявність модуля розпізнавання тексту, і Sikuli не є виключенням. За рахунок API з'являється можливість використовувати такі популярні засоби як TestNG або JUnit а також технологію DDT в поєднанні з можливостями Sikuli, це створює достатньо потужні можливості для тестування програмного інтерфейсу користувача [7].

Тож Sikuli є універсальним, потужним і безкоштовним засобом автоматизації тестування з відкритим вихідним кодом. Головною перевагою цього засобу є наявність досить гнучкого і функціонального API. А певні незручності що присутні в системі не завдають відчутного дискомфорту при автоматизації тестування.

T-Plan Robot, також відомий як VNCRobot, базований на VNC [3](використовує RFB протокол). T-Plan поєднує в собі два основні підходи: базований на зображенні і базований на абсолютних координатах екрану. Це дає повну впевненість в надійності роботи програми незалежно від зміни графічного інтерфейсу [4].

Щодо DDT , його використання є неможливим через відсутність засобу для зчитування зовнішніх даних, файлів і т.д. T-Plan Robot надається у вільному доступі для завантаження, надається згідно з GPL [5]. Але існує також платна Enterprise версія яка надає широкий спектр підтримки цього програмного забезпечення.

Отже T-Plan Robot працює виключно на віддаленому сервері, тож

потребує додаткову машину для встановлення. Його функціональні властивості надають необхідний мінімум для тестування і в той же час не є досить різноманітними та специфічними, що не є його сильною стороною.

Вибір оптимального засобу розпізнавання зображень

Перш за все варто звертати увагу на наявність API, адже робота передбачає виконання практичної частини: написання власної програми яка автоматизує тестування за допомогою саме засобів пошуку графічних елементів. Серед певної кількості API перевага надається мові програмування Java. Серед розглянутих систем тестування сюди підпадає Sikuli.

Для здійснення доступу TCP/IP до віддаленого комп'ютера з будь-якого іншого комп'ютера або мобільного пристрою потрібно встановити програмне забезпечення VNC, яке відображає у вікні вашого комп'ютера весь екран віддаленого комп'ютера та передає йому коди натиснутих клавіш та команди мишки, таким чином надаючи користувачу повний «ефект присутності» [4]. З цим параметром лідирують Ranorex та Sikuli.

Модуль розпізнавання тексту є невід'ємною частиною професійно засобу тестування GUI. Якщо виникає ситуація, за якої, з різних причин, не вдалося знайти зображення на екрані, програма зможе автоматично здійснювати пошук по тексту, і не завершиться помилкою, а скоріш за все знайде шуканий елемент управління, адже їхні назви рідко коли змінюються. Тож сильним модулем для розпізнавання текстів володіють всі з наведених систем крім T-Plan [6].

Всі засоби крім Sikuli надають дуже детальний звіт з усіма можливими даними, та підкріплюють кожен дію відповідним скріншотом. У Sikuli ж ситуація інакша: всі виконані дії логуються із збереженням абсолютних координат зображення або дії на екрані. В загальному випадку для роботи цього цілком достатньо, хоча є можливість вдосконалити існуючу систему звітності у Sikuli для кращого перегляду тестів.

Тепер можна чітко зробити висновок, що для пошуку графічних елементів найоптимальнішим засобом автоматизації тестування є Sikuli. Він поступається місцем іншим засобам автоматизації лише за одним параметром, який є найменш важливим серед усіх наведених метрик.

Література:

1. *Mohammad Rafi Automated Software Testing. A Study of State of Practice / Dudekula Mohammad Rafi & Kiran Moses // School of Computing Blekinge Institute of Technology – Sweden - 2010. – С. 5*
2. *Ranorex .NET Documentation – <http://www.ranorex.com/Documentation/Ranorex/>*
3. *Katam Reddy Picture-Driven Computing / Katam Reddy//Information Inc.,- 2010 - Bethesda, Maryland, USA.*
4. *Kenneth R. Virtual Network Computing/ Tristan Richardson , Kenneth R. Wood and Andy Hopper// Quentin Stafford-Fraser - 1998. – С. 43*
5. *T-Plan Robot Overview Documentation – <http://www.tplan.com/robot/index.html>*
6. *Template matching algorithm – http://docs.opencv.org/doc/tutorials/imgproc/histograms/template_matching/template_matching.html?highlight=matchtemplate*
7. *Normalization in Template matching algorithm – <http://werner.yellowcouch.org/Papers/subimg/index.html>*
8. *SAD – Sum of the Absolute Differences – <https://siddhantahuja.wordpress.com/tag/sum-of-absolute-differences-sad/>*