# Reports on using an ABB robot in Android mobile testing automation

Abda Albuquerque, Alice Castro, Heryck Barbosa
{abda.albuquerque,alice.castro,heryck.barbosa}@sidia.com
Sidia Institute of Science and Technology - SIDIA
Manaus, AM, Brazil

## ABSTRACT

There are many automation available for our daily lives that seek to improve performance and quality ensuring time reduction. For the area of telephone tests, there are many possibilities to automate mainly by dealing with sensors. The purpose of the article is to present a different way of performing the Rotation Vector Vision Crosscheck Rotation Test Case, through an ABB robot that has 7 axis and compare its performance with the existing automation proposal. An experiment of 20 executions was performed on the YuMi® robot for each of the 5 selected models. The result obtained was a rate of 85% accuracy, disregarding one of the models due to a determining factor.

## CCS CONCEPTS

• **Computer systems organization** → **Robotic autonomy**; • **Software and its engineering** → **Operational analysis**.

## KEYWORDS

ABB robots, test automation, software teste, Android, Google Tests, Rotation, mobile

## 1 INTRODUCTION

Automation is becoming increasingly predominant in our daily lives as we look for ways to improve performance, save time, and improve quality. As a result, there is a growing interest in the power of automation in the field of telephony, which offers numerous opportunities for optimization and improvement. According to Shrouf et al. [4], it is possible to optimize processes with minimal human intervention, leading to fewer errors, reduced use of resources, and greater efficiency.

We are presenting an alternative way to automate the Rotation Vector Computer Vision Crosscheck (RV) Test Case for Android Mobile [1]. This test was created to certify, through sensors such as gyroscope, accelerometer, and magnetometer, the ability of a smartphone to detect its orientation in space, evaluating its performance [2]. Figure 1 presents how to manipulate the smartphone to execute the test case around the dot pattern in three different directions: (1), (2), (3) [1].

Our approach involves simulating the same movements that the test requires, and compare the performance of two robotic arms. The first robotic arm created for this specific test case is a three-axis arm that uses an ATmega328p microcontroller and stepper motors. It was designed specifically to meet test case requirements [2]. The second robotic arm, developed by ABB Robotics, has two arms, each with seven axis, but we will be using only one arm for this test case.

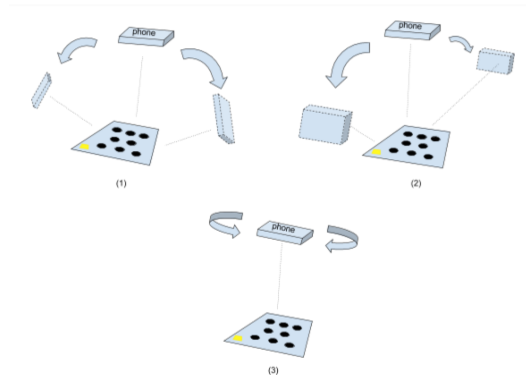



**Figure 1: Manipulating the device under test.[1]**

The present paper is structured in the following manner: Section 2 provides a brief overview of the robotic arms. Section 3 describes the first robotic arm built to this test. Section 4 presents an new proposal to execute RV Test Case. Section 5 describes the experiments and results obtained. Finally, Section 6 concludes the work and outlines future research.

## 2 BACKGROUND

### 2.1 YuMi® Dual-Arm Robot - ABB Robotics

The ABB robot, which is commonly referred to as YuMi®, has two articulated arms, each with 7 axis. It is a safe and efficient collaborative robot that can work alongside humans without wall's protection. YuMi® is designed for small parts assembly and is known for its productivity [3].

YuMi® robots can be programmed using the Robotic Studio software and Rapid Structured Language, developed by ABB for its robots. There is also the ROS language (robot operating system) which is open-source and makes it possible to program YuMi®, but setting up the environment can be complicated, in addition, the Rapid language has an advantage over ROS in terms of abstraction, making it much easier to program even complex tasks.

## 3 RELATED WORK

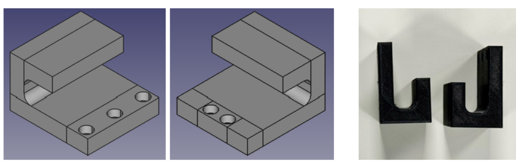
### 3.1 Automating Android Rotation Vector Testing in Google's Compatibility Test Suite Using a Robotic Arm

The authors Albuquerque et al. [2] created a robotic arm using reusable parts to perform the RV test case for the Google Compatibility Test Suite on the Android operating system. Their experiments showed that the automation successfully passed the test case.

During the experiment, eight different smartphone models were tested, each only once. Objective was to point out if in the first attempt to perform the test, the robotic arm was more successful than the human tester. Human testers had a success rate of 37.5% in their first attempt, while the robotic arm reached a success rate of 75%. However, there were some models in which the robotic arm failed. The study authors attributed these flaws to minor technical problems during the installation of smartphones in the smartphone fitting support that did not allow much flexibility or adjustment.

## 4 IMPLEMENTATION

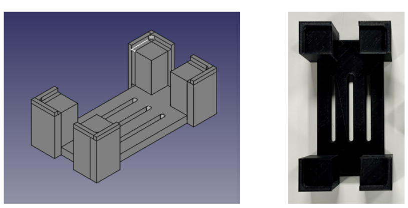
To mitigate possible problems resulting from the different dimensions of smartphones, we have developed a new claw for the YuMi® robot using advanced 3D printing technology. This newly created claw (Figure 2a) can adapt to adjust to any smartphone, regardless of its dimensions (Figure 2b).



(a) (b)

**Figure 2: Claw in 3D printing**

A cradle was specifically designed to enable the claw to grab the smartphone. The cradle has open side spaces (see Figure 3) and was 3D printed using durable material. It has been tested and proven to be compatible with a wide range of smartphone models except foldable models.



(a) (b)

**Figure 3: Cradle in 3D printing**

In Figure 4a, the arm is programmed to reach and grab the smartphone positioned in the cradle. The cradle position is determined in the code. After picking up the smartphone, the arm moves to the next position located on top of the paper with a pattern of dots, the test case implemented at the robot mixes camera usage by focusing in the matrix (Figure 5) with gyroscope sensors. The goal is to fulfill the camera movements with precision checking camera stability.

As programmed, the robot will wait 8 seconds for the tester to start the test case on the smartphone. Once started, the robot moves along the X-axis after a certain period. See Figure 6.

Concluding the previous step, the robot prepares to execute the movement on the Y axis. Unlike the automation proposed by Albuquerque et al. [2], no new external interaction with the arm, and the YuMi® robot can be tilted perfectly with its 7 axis.

(a) (b)

**Figure 4: YuMi® Robot reaching smartphone**

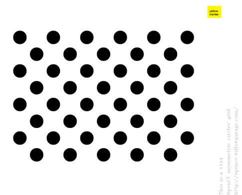


**Figure 5: Pattern of Dots[1]**

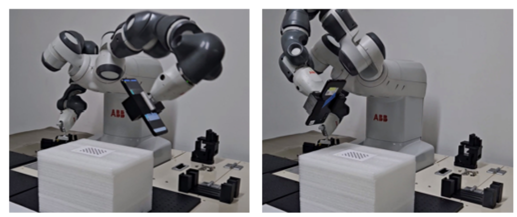


**Figure 6: Inclination X-axis**

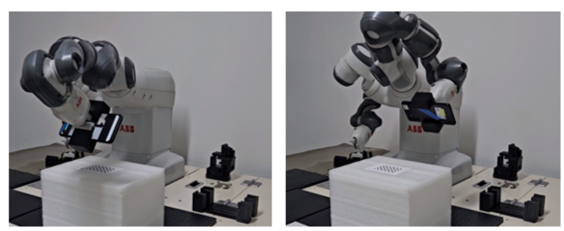


**Figure 7: Inclination Y-axis**

Following to the next step, the z rotation occurs linearly and in moments. When it ends, it goes back to the cradle position to leave the smartphone, while the test case processes the result.

## 5 EXPERIMENTS

We selected the most relevant specifications of 5 models of smartphones, which were tested manually and with the robot. Camera, camera stabilization, and processor are items that we believe are important while performing the test case. It can be observed in the Table 1 that camera stabilization was the only feature almost common to all models.

**Figure 8: Rotation Z-axis**

**Table 1: Model Specifications**

| **Model A** | **Specifications** |
| --- | --- |
| Camera | 50 Mp + 5 Mp + 2 Mp + 2 Mp |
| Stabilization | Digital |
| **Model B** | **Specifications** |
| Camera | 12 Mp + 12 Mp |
| Stabilization | Optical |
| **Model C** | **Specifications** |
| Camera | 108 Mp + 10 Mp + 10 Mp + 12 Mp |
| Stabilization | Optical |
| **Model D** | **Specifications** |
| Camera | 108 Mp + 48 Mp + 12 Mp |
| Stabilization | Optical |
| **Model E** | **Specifications** |
| Camera | 50 MP + 12 MP |
| Stabilization | Optical |

Smartphones have two types of image stabilization: digital and optical. Digital stabilization uses software correction to compensate for unwanted movements during photo or video capture. Optical stabilization uses a gyroscope to detect any irregular movement that may impair image quality by moving the lens's internal elements to make the necessary adjustments.

We experimented with 20 manual executions on each model marking the hits and failures. We were able to analyze that the largest amount of flaws occurred on the Z axis, during the rotation, which should be due to the instability of the tester's hands when moving around the paper, keeping the smartphone always on the shaft. Tests on the same models were done by the YuMi® robot comparing the time of 20 executions on each model and the marking of hits and failures.

## 5.1 Results

We observed that the A model failed due to digital stabilization problems. Quick movements produced blurred images that could not be corrected by the software which generates the observation that models that have digital stabilization cannot be operated by the YuMi® robot.

Table 2 presents the Accuracy Rate (AR) of each model manually and automated by the YuMi® robot. We have an average of 61% AR for manual tests and 68% AR for tests done by YuMi®, but if we consider that model A will not be part of the robot's scope for not having optical stabilization, the accuracy YuMi® Rate would become 85%.

**Table 2: Manual Accuracy Rate vs Robot Accuracy Rate**

| **Models** | **AR Manual** | **AR Robot** |
| --- | --- | --- |
| **Model A** | 35% | 0% |
| **Model B** | 75% | 100% |
| **Model C** | 95% | 95% |
| **Model D** | 55% | 70% |
| **Model E** | 45% | 75% |
| **Total** | **61%** | **68%** |

The robot execution time is 28 seconds, regardless of the model. While in the tests done manually, we took an average of the execution time of different testers and reached the result of approximately 30 seconds.

## 6 CONCLUSION

With this experiment, we can conclude that regarding the difficulty of automation [2], the claw designed for the YuMi® robot can adapt to different models of smartphones. It also has the advantage of having more axis, managing to reach the smartphone in any position near it, and it is only necessary to feed the cradle with new smartphones so that the robot performs the test.

To compare with the other robotic arm, and disregarding model A, the accuracy rate of the first YuMi® attempts was 75%. During our experiment, we found that various models have their cameras in slightly different positions. It caused some problems, such as depending on the model, we needed to manually adjust the paper position to ensure a successful test. We are currently exploring a solution to this problem for future work. Instead of using printed paper, we are considering using a tablet that contains the image so that in case of test failure, through *ADB* commands, retry the test and adjust the tablet image in a new position or size until the test passes.

## ACKNOWLEDGMENTS

The authors are grateful for the support offered by SIDIA R&D Institute in SmartGate project. This work was partially supported by Samsung, using resources of Informatics Law for Western Amazon (Federal Law No. 8.387/1991). Therefore, the present work disclosure is in accordance as foreseen in article No. 39 of number decree 10.521/2020.

## REFERENCES

[1] Android Developers. 2024. Rotation Computer Vision Crosscheck Vector. url = https://source.android.com/compatibility/cts/rotation-vector.

[2] Abda Myrria de Albuquerque, Heryck Barbosa, Pedro Lancellotta, João Santos, and Janislley Sousa. 2023. Automating Android Rotation Vector Testing in Google's Compatibility Test Suite Using a Robotic Arm. In *Proceedings of the 8th Brazilian Symposium on Systematic and Automated Software Testing*. 54–63.

[3] ABB Robotics. 2024. Dual-arm YuMi® - IRB 14000. Available at https://new.abb.com/products/robotics/robots/collaborative-robots/yumi/dual-arm (accessed: 2024/01/31).

[4] F. Shrouf, J. Ordieres, and G. Miragliotta. 2014. Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm. In *2014 IEEE International Conference on Industrial Engineering and Engineering Management*. 697–701. https://doi.org/10.1109/IEEM.2014.7058728