Voice Recognition to Control and Monitor Robotic Arm on IoT by Using Smartphone

Ngoc-Hoang Tran
University of Technology and Education – The University of Danang, Danang, Vietnam
tnhoang@ute.udn.vn

Abstract. Robotic control approaches are extensively studied and developed in recent years, especially in industrial manufacturing. They are explored in various ways from using image-processing to voice-controlled applications. Robot communication techniques are required to develop by three dimensions of production: better quality connection; safer and cheaper. In this context, this paper proposes a protocol of control and monitoring robotics by using a developed technology of the 4.0 industries: Internet of Things (IoT). Control systems by voice recognition using Google Assistant on Smartphone and IoT communication by using Adafruit IO with IFTTT platform as data transfer intermediaries. Our approach applies to control an own developed robotic arm with Arduino Uno as the controller. Our paper shows also the voice controlling commands are successfully transmitted via IoT technology and the desired operations are performed successfully and also analyse its advantages and disadvantages and the perspectives.

1. Introduction
Today, robots take an important role in the science and technology development. Indeed, robots have been widely applied in military, industrial, and agricultural techniques to replace people’s role in dangerous environments. Many technologies are developed for controlling robots in recent years. Among them, Voice recognition technique is proved as an intelligent method that was increasing widely role in this context such as Google Assistant, Alexa, Cortana [1]. Following, a large number of smart/intelligent robot was designed by using wireless communication. They express the human-robot interaction (HRI) term with more productivity, less energy resource waste and more convenience than old control methods by physical communication (keyboard, computer, LAN, etc.). For example, the researches [2,3] uses Bluetooth to communicate/control to robots from the Bluetooth transmitter to command robot to perform operations. These studies use mainly circuit systems with old sensors such as RSC364 (used in radio transmission) to analyse the voice frequency to command. This approach faces disadvantages such as controlling is possible in low distance depends system, high cost for circuit construction and its quality depends on the recording quality of the microphone. Furthermore, [4,5] propose a new Bluetooth transmitter approach via smartphone. The control design in these works was simpler than the old type. By using WIFI, 3G, 4G, or LAN network, communication with the Robot is significantly improved in distance terms, more flexibility and more convenience [6]. However, they are still limited at storage capacity, direct / indirect connection and poor system security. In our part, we propose an advanced approach to control robot by using IoT communication. Voice recognition command through Google Assistant, transmit/feedback IoT by intermediates servers. By
this way, this approach has many important ad-vantages that are more secure and safely; controlling robot is not depend on physical distance; easy to access for monitoring by historical data on server. This article is structured as following: the next section proposes the control/monitor robot process. This section presents also the used techniques in this work. Based on this, we show our results that tested on an own developed robotic arm. The conclusion and the perspective will close the article.

2. Proposed Control and Monitor Process
This process is represented by three main parts: Human - Voice command on Smartphone environment; IoT platform for transmit order and receive feedback from sensor system and robotic arm controlled by Arduino Uno, as described as figure 1.

Figure 1. Proposed process of controlling and monitoring robotic arm by voice command on IoT

2.1. Human – Voice command on Smartphone environment
To convert voice to text or computer commands, it must be performed by a complex process with many steps. Basically, an Analog-to-Digital Converter (ADC) converts these analog (analog) waves (human voice) into data that system can understand, then filtering (separate) noise into different frequency bands to eliminate, see more details in [7]. Fortunately, all of these complex processes are supported by Google Assistant. Google Assistant system will have to receive sound from the human speaking on the phone, the phone will record the monologue through google, the data will be processed as text characters.

2.2. Establishment of IoT platforms
The communication module used in this work was the advanced development protocol, which is a virtual interface platform that communicates with the ESP32 chipset: Adafruit IO. This is an IoT platform built around the Message Queue Telemetry Transport (MQTT) Protocol [8], that has extensive literature focused on IoT solutions, all developed by the Adafruit Industries.
In other side, If-This-Then-That (IFTTT) is a widely used platform with over 11 million users [9] which is considered as a useful bridge between Google Assistant and Adafruit IO platform. IFTTT platform is designed as a trigger-Action platform which support stitching together various online services APIs such that end-users may write simple conditional programs. These simple programs often take the form “IF triggering condition, THEN take a specific action”. We designed the synchronization operation of these two servers. When a voice command is recorded on Google Assistant, it is encoded and transmits this encrypted signal into text form from Google’s API through ITFFF Server to Adafruit IO and sent directly to Arduino ESP 32.

2.3. Robotic Arm controlled by Arduino
Arduino is an open-source platform based on hardware (8-bit AVR Atmel Processor or 32-bit ARM Atmel) and software represented correspond as a simple I/O board (equipped with a USB interface, 6 analog input pins, 14 digital I/O pins allow to expand with various boards) and a Processing/Wiring language developed environment (called Arduino Integrated Development Environment, also Arduino IDE). This Arduino IDE is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries to user easily the AVR C programming language on which it's based. In our paper, the robotic arm controlled by Arduino is designed with 3 Degree of Freedom (DOF) as described in figure 2 and table 1.

![3DOF Robotic Arm design model](image)

**Figure 2.** 3DOF Robotic Arm design model.

| Names       | Descriptions                        |
|-------------|-------------------------------------|
| Overall size| 250 x 60 (cm)                       |
| Material    | Metal                               |
| Processor   | Arduino ESP32 (5V – 90mA)           |
| Motor       | Mini Servo (with feedback signal)   |
| Sensor      | IR Infrared Proximity (5V – 15mA)   |

**Table 1.** Robotic Arm Specifications

3. Results and discussions

Following to proposed process, following steps describe our implementation and its results:

1) Create an applet editor (on ITFFF platform), where we choose triggers ("If This") and the subsequent actions ("Then That"). We choose in the next "Google Assistant" as the service for our trigger, then set up its settings such as: the Google Assistant's response, language, etc. In our case, the phrase "Ok I'm in processing" to response command order. It implies that when the ESP processor is received the request and is performing actions. The final part of this applet is the Action, we choose "Adafruit", meaning to "Send data to Adafruit IO". Till this applet is completed, the converted voice-command to text from Google Assistant will be sent to Adafruit IO platform which transmit to microcontroller. Figure 3 show us an example of voice - command by Google Assistant to ESP processor and its response.

2) Create a new feed of Adafruit IO: In which we create threes "postData" variables correspond to threes joints of Robotic Arm and "onoff" variable to represent feedback values from ESP processor while it is performing.
4

Figure 3. Voice – Command to Robotic Arm by Google Assistant.

In addition, ESP32 processor connect to Adafruit IO by MQTT Protocol. It’s performed by to install the Adafruit MQTT Client library, which can be found under the Arduino Library Manager [10] by including at the start of our program:

```cpp
#include <ESP32-WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
```

Once the connection is completed, ESP processor will receive the voice-command from Google Assistant through above step 1 and Robotic Arm perform actions corresponds. Sensor from mini-servo feedbacks to Adafruit Server to monitor Robotic Arm’s activities. Figure 4 prove us controlling and monitoring Robotic Arm’s performing process.

Figure 4. Interface of monitoring of Robotic Arm’s performance on IoT server.

In figure 4, information feedback from joint A, B, C and onoff correspond to the components of Robotic Arm described in figure 2. They are updated automatically and continuously on Adafruit IO server to monitor in real-time.

3) Finally, these feedbacks are synthesized in our Adafruit server and send a notification on our smartphone when Robotic Arm has finished its activities, as described in figure 5. The results show us that the control and monitoring of the robot's execution are successful. The feedback from the robot’s sensors (~5-10ms/cycle) almost no-delay.
However, this work still has some limitations: (i) Voice recognition is depended on Google Assistant in which it has not specifically designed for each individual language. Moreover, the extension with the effective filters to improve recognition quality is impracticable. (ii) Using two common Adafruit and IFTTT platforms will be difficult to avoid security risks, and its connections are unstable.

4. Conclusions and future works
Our contribution in this article to provide an advanced approach of controlling and monitoring robot's operations. Voice commands are successfully transmitted from Google Assistant on Smartphone through IoT cloud to Arduino processor. User will get notification while desired activities has done successful. This research reduces human activities in dangerous places or situations: such as radioactive mineral exploration, geological measurement. In perspective, an extension monitoring protocol with camera system is necessary to developer on integrating with our methodology. In addition, robot positioning with build self-propelled algorithm make improve supervision system.

5. References
[1] López, Gustavo, Luis Quesada, and Luis A. Guerrero. 2017. Alexa vs. Siri vs. Cortana vs. Google Assistant: a comparison of speech-based natural user interfaces. International Conference on Applied Human Factors and Ergonomics. Springer, Cham.
[2] Abdullahi, Zakariyya Hassan, et al. 2015. Mobile robot voice recognition in control movements. International Journal of Computer Science and Electronics Engineering (IJCSEE) 3.1: 11-16.
[3] Fezari, Mohamed, and Mounir Bousbia-Salah. 2006. A voice command system for autonomous robots’ guidance. 9th IEEE International Workshop on Advanced Motion Control, IEEE.
[4] Kahar, Saliyah, et al. 2011. Data transferring technique for mobile robot controller via mobile technology. International Conference on Pattern Analysis and Intelligence Robotics. Vol. 1. IEEE.
[5] Ersahin, Gonca, and Herman Sedef. 2015. Wireless mobile robot control with tablet computer. Procedia-Social and Behavioral Sciences 195: 2874-2882.
[6] Hassan, Mohammad Ashri Abu. 2012. A review of wireless technology usage for mobile robot controller. Proceeding of the International Conference on System Engineering and Modelling (ICSEM).
[7] Andics, Attila, et al. 2010. Neural mechanisms for voice recognition. Neuroimage 52.4: 1528-1540.
[8] Kodali, R. K., and Kopulwar S. M. 2016. A low-cost implementation of MQTT using ESP8266. 2nd International Conference on Contemporary Computing and Informatics (IC3I). IEEE.
[9] IFTTT, 2011. https://platform.ifttt.com.
[10] Maguluri, Lakshmana Phaneendra, et al. 2018. Efficient Smart Emergency Response System for Fire Hazards using IoT. International Journal of Advanced Computer Science and Applications (IJACSA) 9.1: 314-320.