DEVELOPING ANDROID-BASED DSLR CAMERA WIRELESS CONTROLLER APPLICATION FOR LOW-END DSLR CAMERA

Endang Tirtana[1], Eko Budi Setiawan[2]

[1,2] Informatics Engineering Department,
Faculty of Engineering and Computer Science, Indonesian Computer University
Jalan Dipatiukur 112 – 116 Bandung
Email : endang7tirtana@gmail.com[1], ekobudisetiawan@ymail.com[2]

ABSTRACT

The photography technology development in DSLR (Digital Single Lens Reflex) has triggered vendors to make innovations related to digital camera [1]. One of those beneficial innovations in DLSR camera is that it is able to connect with other mobile devices using wireless network. Its primary function is to control camera from distance and to instantly upload photos to social medial [2]. However, not all DSLR cameras are equipped with this feature. Only the latest mid-end and high-end edition DLSR camera have the feature [1].

In order to solve the previously mentioned issue, the researcher conducted a trial using Arduino and Bluetooth as the camera's wireless network [3]. The result showed that Arduino did not have its own module and, as a result, users would need to add new modules if they wanted to add new features. This was considered ineffective and inefficient. Moreover, Bluetooth connection is rarely used nowadays due to the limited transfer rate and area coverage compare to using WiFi connection [4].

In line with the aforementioned explanation, a research was held with the main objective to access and operate low-end DLSR camera using Raspberry Pi without other additional modules. The Raspberry Pi was operated using access point and controlled by smartphone.

.

Keywords : camera, DSLR, wireless, Raspberry Pi, Android

1. PRELIMINARY

The development technology on DSLR (Digital Single Lens Reflex) triggers the vendors to make innovations related to digital cameras [1]. One feature that is useful when it is a DSLR can be connected with mobile devices using a wireless network. The goal is to be able to control the camera remotely and can instantly upload the photos to social networks [2]. But not all DSLR cameras have these features, only a DSLR camera mid-end to high-end that have the latest edition [1].

Based on the questionnaire the researchers gave to the community of photographers in the Bandung city who have a low-end DSLR cameras, found several problems in the operation and accessing the files on the memory card. Access a memory is still using a data cable or remove the memory card. Another problem is when controlling DSLR camera from the distance, it is limited by data cables that have several meters, so that the user mobility is very limited. Other problems when operating the camera DSLR directly frequent shocks resulting images are not clear. It also often occurs when taking an object with a low shutter speed despite using a tripod [2].

Previous research conducted by Yuris Octa Primariadi using Arduino and Android-based bluetooth. Android can control digital SLR camera and tripod moving in accordance with the wishes of the user. Connections camera and tripod use two bluetooth module RN-42 and HC-05 [4], making it less efficient because they have to keep changing the settings. The module is an additional tool for not embedded directly in Arduino. Lusiana Citra Dewi [3] conducted a study of the development of wireless network technology. He compares the infrared technology, Bluetooth, and WiFi. The conclusion of the study explained that the development of wireless network technology very rapidly led to wireless transmission technology, infrared and bluetooth becoming obsolete. One reason for the limited speed of transmission [3].

Raspberry Pi is a Single Board Computer (SBC) is quite popular although there are several others like BeagleBone SBC, Intel Galileo, PandaBoard, CubieBoard, and others [5]. The latest generation Raspberry Pi is a Raspberry Pi 3 just released in February 2016. The Raspberry Pi board 3 carries shape and the same interface with Raspberry Pi 2 and 1, but increased in System on Chip (SoC) that is 10 x faster than raspberry Pi 1 and has been integrated with wifi and low power bluetooth module [5].

1.1. Identification of Problems

Based on the background of the preliminary problems, there are several obstacles that occur when operating a DSLR camera, either directly or wirelessly using Bluetooth network. These problems researchers formulated as follows.

1. The difficulty of the operation of a DSLR camera wirelessly using Bluetooth network because it has limitations in transmission speed compared with the WiFi network.
2. The difficulty of the operation remotely and access files low-end DSLR camera which is currently still using the data cable so that the photographer had trouble.

2.2. Purpose and objectives

The purpose of this research is to build applications based on Android DSLR camera controllers for low-end DSLR cameras. While the purpose of this research is:

1. Facilitate the operation of a DSLR camera using WiFi or wireless network using Android smartphones.
2. Make it easy to access and download files on the memory card using a WiFi network with the Android smartphone.

2. LITERATURE REVIEW

1.1. Photography

Photography is the medium of communication between photographers and photo lovers. In order for such communications can be delivered properly, the photographer should be able to compose a photo with good. Composition in photography is a way of attaching the visual elements in a photo[5]. Photographer is a profession that cultivate the field of photography.

1.2. Android

Android project began in early 2003 by a small company called Android, Inc., before the term smartphone widely recognized by the current user that has a large touch screen, high speed internet, GPS, and other fun stuff. In 2007 the company Android, Inc., acquired by Google [6]. In October 2012, there were 700,000 applications available on Android, and about 25 million applications have been downloaded from Google Play. A survey in May-June 2013 found that Android is the most popular platform for developers, used by 71% of mobile application developers and on September 1, 2013, 1 billion devices have been activated [7].

Google wants Android to be open and free, and therefore most of the Android code was released under the open source Apache License. This means that anyone who will use Android can do this by downloading the entire source code for Android [7]. The manufacturers of smartphones that use the Android operating system can add additional applications to differentiate their products from other manufacturers. Due to the open nature of Android that many developer community to build applications for Android. The availability of various types of free applications on the Google Play Store makes many people who switch to using a smartphone based on Android.



Figure 1 Android Logo

1.3. Digital Single Lens Reflex (DSLR)

Digital Single Lens Reflex (DSLR) is a digital camera that uses an automatic mirror system and pentaprism or pentamirror to transmit light from the lens to the viewfinder. Viewfinder is a small hole behind the camera to view images of objects [8]. DSLR cameras have advantages in terms of users' freedom to change the camera settings as you wish. These features and the technology applied to digital SLR cameras vary depending on each type. Differences in the technology applied in each differentiated by grade DSLR cameras [3].

1.4. Flask API

Flask API is an implementation of a web API that can be explored using the supplied Django framework REST. REST web framework Django is a Python-based open source. The web applications created with Flask stored in one file .py. Flask is a web framework that is simple but can be expanded with a variety of additional libraries that correspond to the needs of its users. Flask API keep the commandments of gphoto2 library and piggyphoto library.

1.5. Raspberry Pi

Raspberry Pi (RPi) is a mini-sized PC such as a credit card which was developed by the Raspberry Pi Foundation. Initially created for the computer learning process for the students at the school. Along with the development of technology, now RPi can be used for various applications, such as home automation, web server, DNS Server, and others. RPi was first released in February 2012 with a model B Rev.1 with 256 MB of RAM. In September 2012 came out a model B Rev.2 with the addition of RAM to 512 MB. In February 2013 came out the model A but with 256 MB RAM and in April 2014 out variants of the Raspberry Pi Compute Module [9].



Figure 2 Raspberry Pi 3

In 2015 issued a second generation that RPi 2 with a variety of types. Quite striking change in RPi 2 with the addition of 1 GB of RAM so. And the latest generation in February 2016 issued the latest RPi models 3. In the new model existing 802.11n wireless module and low power bluetooth [9].

1.6. Language Programming

The programming language used in the construction of a DSLR camera wireless applications based on Android for low-end DSLR cameras are:

1.6.1. Java

Since it was first released in 1996, Java became a popular programming language for developers around the world [10]. Around the year 2012, 9 million active developers using the Java programming language [10]. Java has been widely accepted by developers as an alternative to the programming language. Most companies that use a lot of Java to meet programming needs. Java is widely used to implement Internet-based applications and software used to communicate over a network [11].

One reason for the programming language Java became famous for platform independence. Which means that Java programs can run on many types of computers. A Java program can run on any computer with the first already installed Java Runtime Environment (JRE). Java programs can run on Windows, Macintosh, Unix, and Linux computers, even a Java program can run on mobile phones [12].

1.6.2. Python

Python is an open source programming language that is extensively used to treat some types of problems in programming. Python is widely used to improve software quality, developer productivity, portability program, and integration components. Python is used by at least hundreds of thousands of developers around the world in areas such as internet scripting, system programming, user interface, product customization, numerical programming, and more [13].

Python was originally developed in the late 1980s at the National Research Institute for Mathematics and Computer Science by Guido van Rossum as a successor to the ABC language [14]. Since it was introduced immediately popular because Python syntax clear and expressive. Python was developed with a focus on ensuring that the Python code readability. Python includes a high-level programming language, which means Python code written using the English language is largely recognized. In contrast to the low-level programming languages such as Assembler is closer to how computers think but poorly understood by humans without experience.

Because the syntax is clear and a high level programming language Python makes a very valuable for anyone who wants to learn programming. Because of the Raspberry Pi Foundation recommends Python as the programming language.

1.6.3. Gphoto2 Library

Gphoto2 is a Linux-based software that is used to control a digital camera. Currently gphoto2 own hundreds of digital camera drivers who are in the market, so it can be detected well in Linux [15]. Gphoto2 is part of the open source Linux for use on a digital camera. Gphoto2 developed under the GNU LGPL license, because it is also available for commercial use [16].



Figure 3 Gphoto2 Logo

1.6.4. Libgphoto2

Libgphoto2 (gphoto2 library) contains the commands that can be used to control a digital camera. Libgphoto2 a core library designed to allow access to a digital camera with an external program. Figure 3.15 shows the global architecture of Libgphoto2.

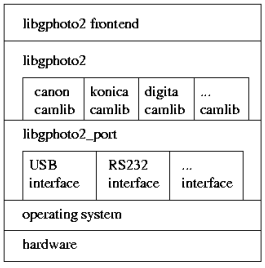


Figure 4 LibGphoto2 Architecture

Communication port and protocol enables the camera to create a complete modularity. To support the physical layer (physical layer), such as IEEE1394 just add a new port to libgphoto2_port. To support new models of digital cameras only need to provide new camlib by calling the necessary commands. Commands from the gphoto2 library Python-based programming summarized in a new book called Piggyphoto. Piggyphoto can be downloaded by typing the command sudo apt-get install piggyphoto in terminal Raspberry Pi. Basic command of Piggyphoto used in this study include:

a. Preview.py

Preview.py is a command to control the DSLR camera screen from a distance and will be displayed on the screen Raspberry Pi if the command is typed at the terminal.

b. Snap.py

Snap.py an order to take a picture that has been displayed on the screen, the command is typed in the terminal Raspberry Pi.

c. Focus-snap.py

Focus-snap.py gives commands DSLR camera to take pictures automatically with the first turn focus to lock an object that caught the camera lens.

Defining the abbreviation and acronym done early in the script either in the chapter or in abstractions. Furthermore, abbreviation and abstractions do not need to re difinisikan. Common abbreviation like IEEE does not need to be defined. Should not use the abbreviation form in writing the manuscript title.

3. DESIGN

3.1. System Overview

A general description of the application to be made can be seen in Figure 5 below.

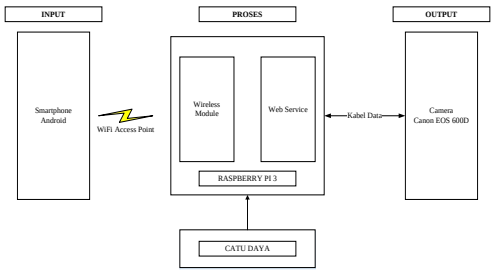


Figure 5 A Diagram Block System

In general, how the system works DSLR camera controlling device Wireless can be explained as follows: the input coming from the user (users) who use Android smartphones and connected to WiFi. Once connected to the access point Raspberry Pi, Android smartphone communicating via Restfull API and access the commands that are already stored in the Raspberry Pi. The commands are used to control focus, ISO, aperture, shutter speed, live view, capture, and download files in the camera memory. After receiving input from the Android smartphone, Raspberry Pi execute the command to run the DSLR camera

.

3.2. Analysis System Design

System that will be built is a DSLR camera wireless controller application based on Android for low-end DSLR cameras. The system to be built can be seen in Figure 6 below.

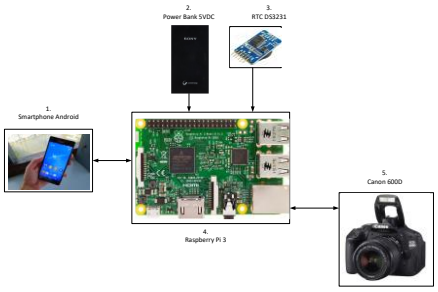


Figure 6 Concept System to beb Built

Raspberry Pi serves as processing orders submitted by the Android smartphone. Android Smartphone Raspberry Pi gives commands to enable features that exist on DSLR cameras using WiFi or wireless network. After setting the appropriate application users simply press the snap button to take a picture, if the picture is taken is not appropriate to stay the process is repeated from the beginning. Storage time at Raspberry Pi not run in real time if it is not connected to the Internet, because it added DS3231 RTC module.

3.3. Analysis System Configuration

The system configuration of an application made is divided into several stages which will be described below.

3.3.1. Network Configuration

For the device to be connected via WiFi networks without requiring an internet connection is required several stages. The first stage of configuring the DHCP server and set Raspberry Pi into a static IP so that when connected to the device does not require IP configuration reset. In this section to configure the access point, this step is necessary in order Raspberry Pi can transmit WiFi signals automatically. For the first Access Point configuration change into a static IP settings by typing:

```
$ sudo nano /etc/network/interfaces
```

Figure 7 Enter to Network Settings

Change Raspberry Pi IP into static as in figure 8.

```
iface wlan0 inet static
address 192.168.10.1
netmask 255.255.255.0
```

Figure 8 Static IP Configuration

To configure access point then create a file by going to:

```
$ sudo nano /etc/hostapd/hostapd.conf
```

Figure 9 Access Point Configuration

Then create a file like the figure 10.

```
interface=wlan0
ssid=RPi_cam
country_code=US
hw_mode=g
channel=6
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=raspberry
wpa_key_mgmt=WPA-PSK
wpa_pairwise=CCMP
wpa_group_rekey=86400
ieee80211n=1
wme_enabled=1
```

Figure 10 Access Point Setting

3.3.2. Gphoto2 Library Configuration

Gphoto2 library can be downloaded for free by typing the command sudo apt-get install gphoto2 on Raspberry Pi terminal. Election library must be adapted to the type of DSLR cameras because not all command gphoto2 library can be used.

To install gphoto2 then download these files by typing the following command*:*

```
$ sudo apt install gphoto2
```

Figure 11 Download and Install Gphoto2

Figure 12 are the commands available in the gphoto2 library.

```
        [-m|--mkdir=DIRNAME] [-r|--rmdir=DIRNAME] [-n|--num-files]
        [-p|--get-file=RANGE] [-P|--get-all-files] [-t|--get-thumbnail=RANGE]
        [-T|--get-all-thumbnails] [--get-metadata=RANGE] [--get-all-metadata]
        [--upload-metadata=STRING] [--get-raw-data=RANGE]
        [--get-all-raw-data] [--get-audio-data=RANGE] [--get-all-audio-data]
        [-d|--delete-file=RANGE] [-D|--delete-all-files]
        [-u|--upload-file=FILENAME] [--filename=FILENAME_PATTERN]
        [-f|--folder=FOLDER] [-R|--recurse] [--no-recurse] [--new]
        [--force-overwrite] [--skip-existing]
pi@raspberrypi:~ $ gphoto2
Usage: gphoto2 [-?qvalLnPTDR] [-?|--help] [--usage] [--debug]
        [--debug-loglevel=STRING] [--debug-logfile=FILENAME] [-q|--quiet]
        [--hook-script=FILENAME] [--stdout] [--stdout-size] [--auto-detect]
        [--show-exif=STRING] [--show-info=STRING] [--summary] [--manual]
        [--about] [--storage-info] [--shell] [-v|--version]
        [--list-cameras] [--list-ports] [-a|--abilities] [--port=FILENAME]
        [--speed=SPEED] [--camera=MODEL] [--usbid=USBIDs] [--list-config]
        [--list-all-config] [--get-config=STRING] [--set-config=STRING]
        [--set-config-index=STRING] [--set-config-value=STRING] [--reset]
        [--keep] [--keep-raw] [--no-keep]
        [--wait-event=COUNT, SECONDS, MILLISECONDS or MATCHSTRING]
        [--wait-event-and-download=COUNT, SECONDS, MILLISECONDS or MATCHSTRING]
        [--capture-preview] [--show-preview] [-B|--bulb=SECONDS]
        [-F|--frames=COUNT] [-I|--interval=SECONDS] [--reset-interval]
        [--capture-image] [--trigger-capture] [--capture-image-and-download]
        [--capture-movie=COUNT or SECONDS] [--capture-sound]
        [--capture-tethered=COUNT, SECONDS, MILLISECONDS or MATCHSTRING]
        [--trigger-capture] [-l|--list-folders] [-L|--list-files]
```

Figure 12 Gphoto2 Library

Gphoto2 libraries used for this research to the ISO settings, aperture, and shutter speed.

3.3.3. ISO

The command used for the ISO settings there are two commands, namely to see the value of the ISO and the command to change the ISO value. To view the ISO value can type the command: *gphoto2 --get-config iso*.

```
pi@raspberrypi:~ $ gphoto2 --get-config iso
Label: ISO Speed
Type: RADIO
Current: 400
Choice: 0 Auto
Choice: 1 100
Choice: 2 200
Choice: 3 400
Choice: 4 800
Choice: 5 1600
Choice: 6 3200
Choice: 7 6400
Choice: 8 12800
```

Figure 13 List of ISO Parameter

To change the parameters of ISO users just type the command: *gphoto2 –set-config iso = x,* where x = the value of the ISO parameter in figure 13 By typing the command in figure 14, the ISO value will change in DSLR cameras.

```
pi@raspberrypi:~ $ gphoto2 --set-config iso=200
```

Figure 14 Changing Value of ISO Parameter

3.3.4. Aperture

To change the aperture value of the camera, which used the same command with the command used to change the ISO value. To view the parameter values aperture users simply typing: *gphoto2 --get-config aperture*, as shown in Figure 15.

```
pi@raspberrypi:~ $ gphoto2 --get-config aperture
Label: Aperture
Type: RADIO
Current: 5.6
Choice: 0 4
Choice: 1 4.5
Choice: 2 5
Choice: 3 5.6
Choice: 4 6.3
Choice: 5 7.1
Choice: 6 8
Choice: 7 9
Choice: 8 10
Choice: 9 11
Choice: 10 13
Choice: 11 14
Choice: 12 16
Choice: 13 18
Choice: 14 20
Choice: 15 22
Choice: 16 25
```

Figure 15 Aperture Parameter

To change the aperture value then use the command: *gphoto2 –set-config aperture = x*, where x = the value of the aperture figure 15. Figure 16 is a command to change the aperture value..

```
pi@raspberrypi:~ $ gphoto2 --set-config aperture=5.6
```

Figure 16 Changing Aperture Value

3.3.5. Shutter Speed

To see the shutter speed parameter value by typing *gphoto2 –get-config shutterspeed*.

```
pi@raspberrypi:~ $ gphoto2 --get-config shutterspeed
Label: Shutter Speed
Type: RADIO
Current: 1/4
Choice: 0 bulb
Choice: 1 30
Choice: 2 25
Choice: 3 20
Choice: 4 15
Choice: 5 13
Choice: 6 10
Choice: 7 8
Choice: 8 6
Choice: 9 5
Choice: 10 4
Choice: 11 3.2
Choice: 12 2.5
Choice: 13 2
Choice: 14 1.6
Choice: 15 1.3
Choice: 16 1
Choice: 17 0.8
Choice: 18 0.6
Choice: 19 0.5
```

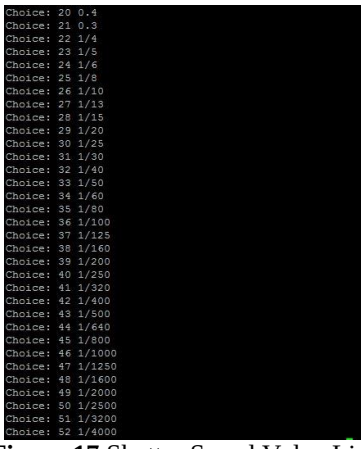


Figure 17 Shutter Speed Value List

To change the shutter speed value then type the command as in figure 17., *gphoto2 --set-config shutterspeed = x*, where x = the value of the shutter speed figure 18.

```
pi@raspberrypi:~ $ gphoto2 --set-config shutterspeed=1/6
```

Figure 18 Changing Shutter Speed Value

To get piggyphoto users can get in the repository https://github.com , by type git clone command, once downloaded will be stored in the folder */home/pi/piggyphoto*.

```
pi@raspberrypi:~ $ git clone git://github.com/alexdu/piggyphoto.git
```

Figure 19 Download Piggyphoto

The command used for this research has been provided by piggyphoto library. Commands can be seen in Figure 20.

```
pi@raspberrypi:~/piggyphoto $ ls
autodetect.py  list-config.py  piggyphoto   sflask.py
bulb.py        list-files.py   preview.jpg  snap.jpg
focus.py       mat             preview.py   snap.py
focus-snap.py  my-logfile.txt  README.md    templates
```

Figure 20 Piggyphoto Command List Library

Commands used in this research is preview.py and snap.py.

3.3.6. Preview.py

Preview.py is a command to control the camera wirelessly liveview. To access this command through the terminal must be entered beforehand piggyphoto directory by typing: *~ $ cd piggyphoto*.

After signing in, type: *sudo python preview.py* then liveview camera can be viewed on the screen raspberry pi..

```
pi@raspberrypi:~ $ cd piggyphoto/
pi@raspberrypi:~/piggyphoto $ sudo python preview.py
```

Figure 21 Liveview Command

Raspberry pi-screen display can be seen in Figure 22.



Figure 22 Liveview Display

3.3.7. Snap.py

Snap.py is an order to take pictures and display it on the screen Raspberry Pi. Type the following command: *sudo python snap.py* on piggyphoto directory.

```
pi@raspberrypi:~/piggyphoto $ sudo python snap.py
Model : Canon EOS 1200D
Status : 0
Port : 4
Operations : 57
File Operations : 10
Folder Operations : 14
USB (vendor/product) : 0x4a9/0x327f
USB class : 0x0/0x0/0x0
Library : /usr/local/lib/libgphoto2/2.5.10/ptp2
Id : PTP
```

Figure 23 Snap.py Command

3.3.8. Flask API Configuration

Flask API serves as a bridge between the Raspberry Pi and Android smartphones. To get a micro framework Flask, type the following command:

```
pi@raspberrypi:~ $ sudo pip install Flask
```

Figure 24 Install Flask

After the installation is completed Flask Micro Framework followed by setting the commands of gphoto2 library and piggyphoto on Flask API that can be accessed via the HTTP-Request.

4. IMPLEMENTATION

The interface is an important part of the application. Interface implementation of applications that are built can be seen below.

4.1. Main Menu Interface

Main menu interface contains liveview, stop liveview, snap, ISO Setting, Setting Aperture, Shutter Speed Setting, Update Profile, and download.





Figure 25 The Main Menu Interface

4.2. ISO Setting Interface



Figure 26 ISO Setting Interface

4.3. Aperture Setting Interface

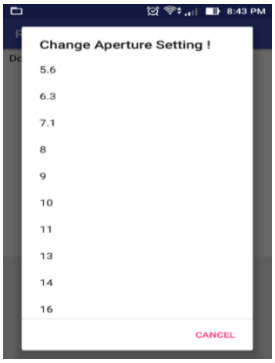


Figure 27 Aperture Setting Interface

4.4. Shutter Speed Setting Interface

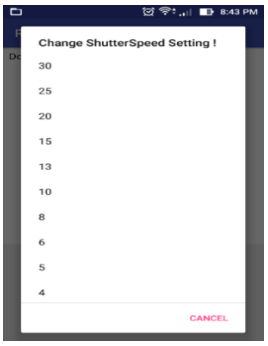


Figure 28 Shutter Speed Setting Interface

4.5. Download Interface





Figure 29 Download Interface

5. CONCLUSION

Based on the results of research, analysis, system design, implementation and testing of the questionnaire, it can be concluded and suggestions for improvement as follows:

5.1. Conclusion

a. DSLR camera wireless control application based on Android can help the operation of the camera.
b. DSLR camera control application based wireless android helps facilitate data access the camera if you want to download.

5.2. Suggestion

This paper are still many shortcomings. It is therefore necessary for the development and further refinement. As for some advice that may be applied to further development are:

a. Completion of the process liveview and snap to avoid conflicts in the camera port so there is no longer an error when connected.
b. Completion liveview camera to the video quality more stable.

BIBLIOGRAPHY

[1] P. Nugrahajati and E. Targo, *Buku Pintar Fotografi dengan Kamera DSLR*, 1st ed. Yogyakarta: Indonesia Tera, 2011.

[2] D. D. Busch, *Mastering Digital SLR Photography*, 3rd ed. Boston: Course Technology, 2012.

[3] Y. O. Primariadi, E. Susanto, and U. Sunarya, "Perancangan Kendali pada Tripod dan Kamera DSLR Menggunakan Komunikasi Bluetooth Berbasis Aplikasi Android," *e-Proceeding Eng.*, vol. 2, no. 2, pp. 2, 4–5, 2015.

[4] L. C. Dewi, "Wireless Technology Development: History, Now, and Then," *ComTech*, vol. 2, no. 9, pp. 1228–1240, 1238–1240, 2011.

[5] G. Andreas and C. R. David, *Professional Android Open Accessory Programming with Arduino*. Indiana: John Wiley & Sons, Inc, 2013.

[6] S. Salbino, *Buku Pintar Gadget Android untuk Pemula*, 1st ed. Jakarta: Kunci Komunikasi, 2014.

[7] W.-M. Lee, *Beginning Android 4 Application Development*, 4th ed. Indianapolis: John Wiley & Sons, Inc, 2012.

[8] D. Indrajit, *Mudah dan Aktif Belajar Fisika*. Bandung: Setia Purna Inves, 2007.

[9] D. Kurniawan, *Membangun Aplikasi Elektronika dengan Raspberry Pi 2 dan WhatsApp*, 1st ed. Jakarta: PT. Elex Media Komputindo, 2016.

[10] P. Sarang, *Java Programming*. New York: Oracle Press, 2012.

[11] P. Deitel and H. Deitel, *Java How to Program*, 10th ed. New Jersey: Pearson, 2015.
[12] D. Lowe, *Java All-in-One For Dummies*, 4th ed. Hoboken: John Wiley & Sons, Inc, 2014.
[13] M. L. T. Cossio, L. F. Giesen, and G. Araya, *Programming Python*, 4th ed., vol. XXXIII, no. 2. Sebastopol: O'Reilly, 2012.
[14] G. Halfacree and E. Upton, *Raspberry Pi User Guide*, 1st ed. West Sussex: John Wiley & Sons, Inc, 2012.
[15] Supriyanto, "Aplikasi Kamera Digital di Linux," pp. 40–43, 2006.
[16] T. Koutecký, D. Paloušek, and J. Brandejs, "Method of photogrammetric measurement automation using TRITOP system and industrial robot," *Optik (Stuttg).*, vol. 124, no. 18, pp. 3705–3709, 2013.