연구논문/작품 최종보고서

2018 학년도 제 2 학기

제목 : A Leap Motion Drone Hand Gesture Control System for DJI Mavic Air

하지크 (인) (학번: 2015310095)

2018 년 11 월 6 일

지도교수: 추 현 승 서명

계획(10)	주제(20)	개념(20)	상세(30)	보고서(20)	총점(100)
8	18	18	28	18	90

* 지도교수가 평가결과 기재

■ 요약

This is a study on controlling a drone (DJI Mavic Air) with much simpler hand gesture control using Leap Motion device. Four components that will be involved are Mac Book pro, Leap Motion device, Android device, and DJI Mavic Air. All of this components are connected through USB, Bluetooth Low Energy (BLE), and Wi-Fi technology. This study main purpose is to develop an easier drone control system that is simple but do not neglect the reliability of a drone and the safety of a user. A lot of advantages can be archived such as even a drone amateur user can easily fly a drone without going through the learning hardships.

1. 서론

가) 제안 배경 및 필요성

Even in the 21st century, gestures play a big role in human society. Example of gestures would be as good as a 'thumbs up' or as bad as cursing with middle finger. Each gesture has a meaning which is the reaction that needs to be followed afterwards. The way we perform gesture also matter for people to understand the meaning behind it.

Drones are free flying devices that do not require pilots being on board. It is a flight control device that can be controlled by a physical controller from a person which is on the ground. A physical controller will be a thing such as a remote control using radio waves, or an artificial intelligence controller that can autonomously fly a drone using a pre-registered programs [1]. These drone control method can be used prior to its study fields and are becoming a hot topic for research. Drone is use in various fields, and for example, in China, drones are being used to monitor cheating in trials. The Gao Kao, which is similar to Korea's SAT exam uses drones to capture various kinds of injustices during examination period [2]. In addition, the unmanned aerial camera of the drone has been used to shoot pictures of places with different kind of environments such as mountainous areas, forested areas, possible natural disasters area as well as accidents occurred on the road. In recent years, the US Amazon has been using drones for courier services within a distance of 400m. Drone technologies which at first were developed for the military purposes such as to observe and surveil the enemy, nowadays have been studied in various aspects. In this way, drone technologies has undeniably changed people's lives in the good way.

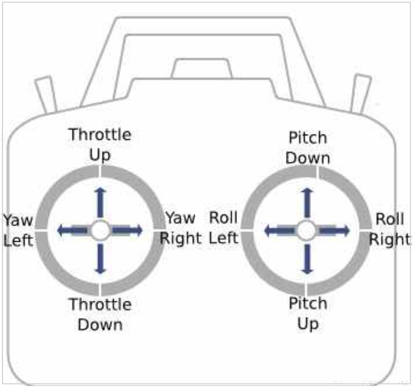

Figure 1. Drone controller diagram

However, the reality is that there are very few users compared to the versatility of the use advantages of a drone described above. The reason is that controlling a drone is difficult for beginners who are new to drone. Firstly, to control a drone, a controller such as a smart phone or a joystick will be needed. As shown in Figure. 1, the control part is divided into two, the left and right part. The left part is for controlling the altitude and rotation of a drone. While the right part is for controlling the drone’s vertical movement which are moving forward, backward, leftwards and rightwards. In addition, the basic drone movement are pitch (move forward and backwards), roll (move leftwards and rightwards), yaw (rotate drone clockwise or counter clockwise), and throttle (move drone up or down). Drone amateur user will most likely be felt difficult to control a drone. Furthermore, to control a drone, a user will have to check the drone’s point of view which will make a drone amateur user feels uncomfortable due to confusion between him or her point of view. Experts who are confident enough to control a drone also need some time to adjust their mind to avoid confusion between the point of view of them and the drone. Moreover, as you can see in Figure 1, most of the time both hands are needed in order to use the drone controller. Thus, it is almost impossible for a handicap person, whom a person who have no arm or cannot do arm movement to control a drone compare to normal person. Therefore, this work suggests control of the drone using Leap Motion.

Figure 2. Leap Motion Controller

Leap Motion is an input device that can be operated by a hand gesture. It is sold by Leap Motion in 2012, and it is a tactile type system that can be operated without touching a mouse or a screen directly, and can be intuitively operated by a hand gesture. It is 200 times more accurate than any previous 3D input device, and it recognizes motion at intervals of 1/100 mm. Leap Motion consists of two infrared cameras and infrared direct LEDs. It records the position of the hand or finger in the 3D space using image analysis by taking two infrared cameras to photograph the hand or finger illuminated by the infrared LED. As Leap Motion can capture the movement of a hand or a finger three-dimensionally, it also can capture hand gesture movements such as moving hand up and down, left and right or back and forth [3]. Therefore, if drone is being control by using this hand gesture recognition device, it will be possible to control a drone at a more intuitive point of view. This will provide a more convenient operation for the user especially drone amateur users.

DJI Mavic Air, a drone that already has a functionality for gesture control but there are certain disadvantages which can be found. The main disadvantages that is crucial is in order to control the drone ones must stand in front of the drone's front camera. This limits the range of controlling the drone by gesture, which is not good for some cases that needed the drone to be controlled by a further distance. On the study further implementation Wi-Fi technology will be used to control the DJI Mavic. This will also restrict the distance compare to the range of DJI Mavic Air remote controller that can cover up to 4000m of range [4], but still better comparing to the method on controlling the drone using gesture on camera. In the future the study will use better connection technology than conventional Wi-Fi to increase the drone control distance, and while gesture is enable a better system that is simple, reliable, safe and secure can be achieved. The development was done by using Leap Motion SDK and DJI Mobile SDK.

나) 졸업 작품/논문의 목표

This project uses Leap Motion controller, a hand gesture recognition device which implement a 3D virtual coordinate system as Figure 3 stated. By using this coordinate system we can implement any algorithm to recognize certain gesture, and do something afterwards.

Relative to this study, we created a system that contains algorithm so that some gesture will be related to certain definition, therefore will be interpreted into certain action. Gesture defined are hand movement that relate to basic drone movement of a drone such as pitch, roll, yaw and throttle, while the action will be the drone movement itself.

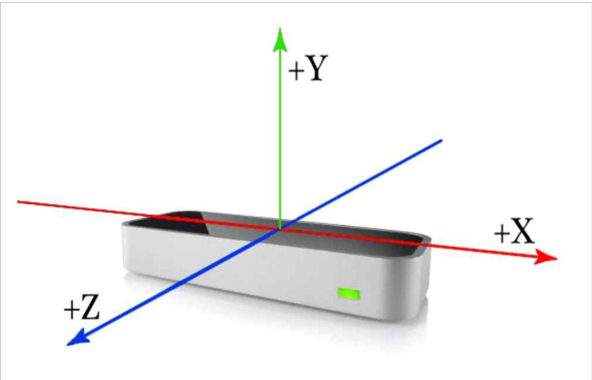

Figure 3. Leap Motion right-handed coordinate system

다) 졸업 작품/논문의 Overview

The study is about controlling DJI Mavic drone using Leap Motion, a hand gesture recognition device. This report consists of introduction, related research, introduction of proposed works, analysis of implementation and results, and conclusion. The related research section introduces the research that influenced this study. The introduction of the proposed work introduces the configuration and implementation method of the system in this study. Implementation and results analysis describes the implemented steps, its analysis, and their future development.

2. 관련연구

All of the related work that will be discuss has the similarity on the study goal, which is using a motion controller and trying to develop a system to allow much simpler drone controls. The difference of this study and others is that the study uses DJI drone. Other study uses AR-drone 2.0 by Parrot which is has less reliability and mobility compare to DJI Mavic air. This study also implement much simpler gestures compare to others.

가) Easy Drone Control System Using Motion Recognition [5]

The control of the drone is made up of pitch, roll, yaw and throttle, and is divided into methods of controlling the direction and movement of the drones, as shown in Figure 4. When a command such as “move_forward” is sent to the drone, it moves based on the point of the drone rather than the point of the user. This may cause a sense of divergence due to the difference in gaze and may cause inconvenience to the user. Therefore, in this paper, to solve this problem, it is necessary to match the direction of the drone with simpler system design such as hand gesture movements.

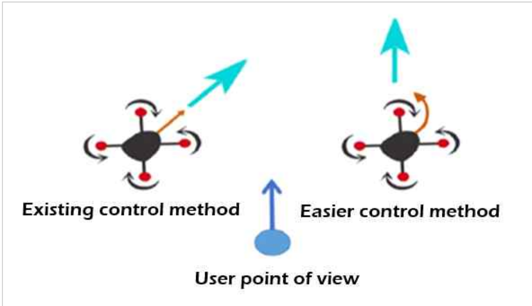

Figure 4. The difference between user point of view and drone point of view

나) Web-based Interactive Drone Control Using Hand Gesture [6]

To learn on how to operate a conventional controller, the person who control the drone first need to know which buttons are present, and what roles those buttons play. This will be very difficult for beginners, and those who want to get a drone license will also have to spend a lot of effort and time. Therefore, in this paper, they propose a method to control the drone by using hand gestures.

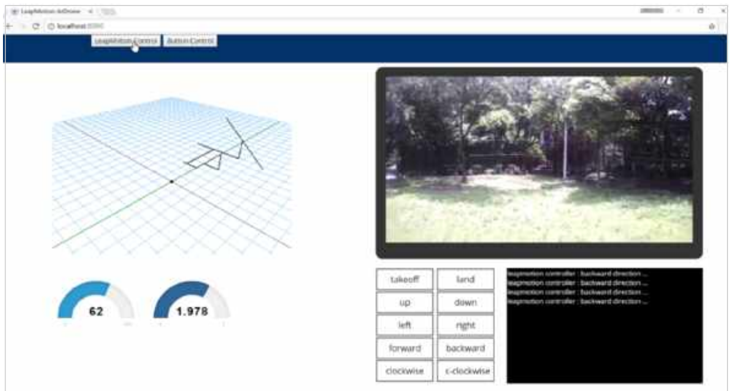

Figure 5. Web-based drone control NUI

The implemented method is the development of a web based NUI (Natural User Interface) system, as stated in Figure 5. A user can communicate with the drone through the NUI without using a physical controller, and the communication is implemented using Wi-Fi technology. Because Leap Motion recognizes a gesture taken by a person, using web-sockets Leap Motion can send a specific command to the NUI and the

NUI can send those specific commands to the drone. Direction and speed of drone are determined by the direction and speed of user’s hand gesture on the Leap Motion device. Therefore, a user can control the drone much simpler.

다) Gesture Control of Drone using a Motion Controller [7]

This research paper presents study about implementation of using a motion controller to control the movement of a drone via simple human gestures. This study has used the Leap Motion device as the motion controller and the AR Drone by Parrot for the drone in the system implementation. The Parrot AR Drone is an off the shelf quad rotor having an on board Wi-Fi system. The AR Drone is connected to the ground station via Wi-Fi and the Leap Motion is connected to the ground station via USB port. The Leap Motion Controller recognizes the hand gestures and relays it on to the ground station. The ground station runs ROS (Robot Operating System) in Linux which is used as the platform for this implementation. Python is the programming language used for interaction with the AR Drone in order to convey the simple hand gestures. Also in the implementation this study uses python codes to interpret the hand gestures captured by the Leap Motion controller, and transmit them in order to control the movement of the AR Drone via the user's hand gestures.

라) Design of Mobile Application for Virtual Reality-based Tour Information Using Leap-Motion and Unity 3D [8]

This paper proposes a mobile application for virtual reality (VR) based tour information using Leap Motion and Unity 3D. The proposed application searches the city using the drone’s first person view camera and selects buildings of interest to provide building information and related videos using 360 VR feature. For stable and intuitive Leap-motion based drone manipulation, this study proposes a drone control algorithm based on DAR (Driving available region). The user's immersion feeling was maximized by providing buildings and roads by topographical authoring tool and asset store of Unity 3D, thus giving the 360 VR of the buildings of interest. As a result, it could be confirmed that the proposed application can effectively provide tourist information with the configuration of the virtual reality environment and the stable drone movement control using the Leap Motion device.

마) Natural User Interfaces for Human-Drone Multi-Modal Interaction [9]

In this paper, a Graphical User Interface (GUI) and several Natural User Interface (NUI) methods are studied and implemented, along with computer vision techniques in a single software framework for aerial robotics called Aerostack which allows for intuitive and natural human-quadrotor interaction in indoor GPS-denied environments. These strategies include speech, body position, hand gesture and visual marker interactions used to directly command tasks to the drone. The NUIs presented are based on devices like the Leap Motion Controller, microphones and small size monocular on-board cameras which are unnoticeable to the user. The research will allow a user to choose the most intuitive and effective type of interaction for their application. Additionally, the strategies proposed allow for multi-modal interaction between multiple users and the drone by being able to integrate several of these interfaces in one single application as is shown in various real flight experiments performed with non-expert users.

바) Motion in the Field, A Study of Movement in Computer Science [10]

The study uses a system that combines the use of a motion controller to obtain information about physical motion of a user with the transmission of outputs to a quadcopter that interacts with the physical environment. In the manual mode, the user sends commands to the quadcopter using hand controls. An existing system is being utilized but increased usability to prevent accidental flight and created commands that are more intuitive for the user. In the automatic mode, the user gives a single command to the motion controller and it runs the A* path planning algorithm over a representation of a known environment to make the quadcopter find and execute the shortest path to its goal.

3. 제안 작품 소개

This research focus on the idea of developing a hand gesture control system to control a drone which make drone control ways more simpler and reliable.

가) 시스템 구성

Basically, there are four component involved which are the Leap Motion device, drone (DJI Mavic Air), notebook (Mac Book pro), and Android device (Samsung S8) as stated in Figure 6. The environment involved for development are Node.js, Android studio, Visual studio code, and Mac OS.

Figure 6. All four components

Basically other related work connect the drone directly to notebook using Wi-Fi socket but because DJI do not provide SDK for Mac OS, one more extra component, the Android device was added. On the other hand, Leap Motion also does not have the SDK that can support Android devices. Therefore, the uses of Bluetooth Low Energy (BLE) as extra connection was needed to connect the whole system. Thus, the system uses Universal Serial Bus (USB), Wi-Fi, and Bluetooth Low Energy (BLE) to enable connection for the whole system.

나) 구현 방식

Figure 7 show the system architecture that involve four components. The implementation of the system varies between each component.

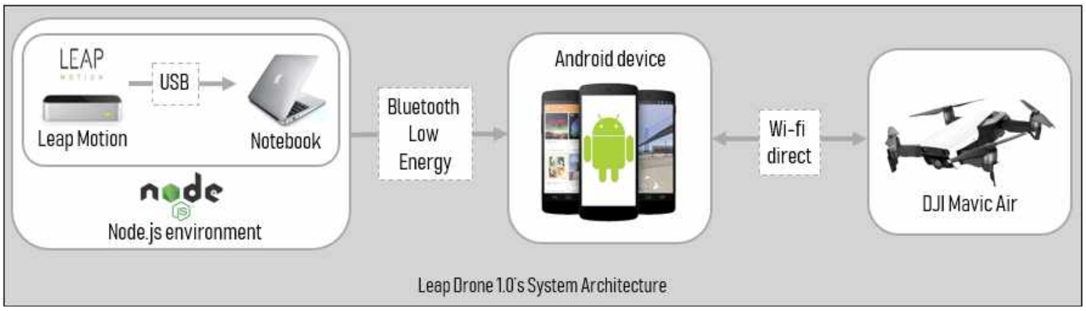

Figure 7. System architecture

The first two component involve are Leap Motion device and notebook (Mac Book pro). The development of the first two component is done on Node.js environment. Leap Motion device and notebook are connected through the Universal Serial Bus (USB). The Leap Motion device will recognize user's hand gesture through its state of the art optical sensor. The recognized user's hand will be represented as vectors in 3D virtual space. This vectors can be manipulated to define a specific gesture that will be given a meaning relative to the system requirement. To get the data of recognized hand, the system used Leap Motion JavaScript based application programming interface (API) that is supported by the Node.js environment.

The third component is the android device (Samsung S8), and obviously involve the Android Studio environment. This component was added because of some restriction discussed previously on system configuration. Bluetooth Low Energy (BLE) was used to connect the first two component with the Android device. The purpose of using BLE is that it has lower energy consumption and more flexibility in terms of protocol compared to the conventional Bluetooth protocol. The Android application that was developed using the API provided by DJI will be responsible to tell the drone about what to do depending on the data received from the first two component.

The fourth and last component is the drone (DJI Mavic Air). Here, the drone will react based on the task triggered by the Android application which depends on the data send through the Wi-Fi socket. The Android application and drone is considered to be a two-way connection. Whenever android application send a command to the drone, an acknowledgment message must be send to the Android application by the drone to indicate that the task was successfully performed or in another word the function code was successfully called on the Android application.

4. 구현 및 결과 분석

The goal of the research is solely to control drone using hand gesture. In order for this to happen, the action or gesture that is being done on leap motion needs to be mapped to the drone movement. In addition if a gesture to move drone forward present, a function call to move the drone forward must also be followed afterwards. The performance of the experiment can be evaluated based on the latency. The latency for this research is the time difference when a user do gesture on Leap Motion device and the acknowledgment from drone indicating function successfully called.

가) 구현

The first implementation needed is between the Leap Motion device and notebook (Mac Book pro). Leap motion recognized user's hand and store the data as a vectors in 3D virtual space. It uses the right hand rule 3D virtual system as shown on Figure 8. The recognized hand data differ in each frame. Thus, by basic vector scale, rotation, and translation it is possible to calculate variables such as hand count, type, position, velocity, and angle. From the hand class, finger class can be extracted which will give variables such as finger count, type, position, and etc. So, the implementation of the first two component begin with the function loop as shown on Figure 9 that can return the hand class data from the frame class for each frame period.

Further implementation will be based on an action map table as Figure 10 which can be refer to check the mapping of a gesture and the drone movement. In addition to the action map in order for this study to achieve its goal on developing a simpler drone gesture control system, it is important to match the gesture done on Leap Motion coordinate system as Figure 8 with the drone coordinate system in figure 11.

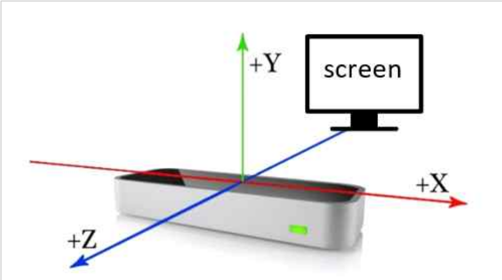

Figure 8. Leap Motion 3D virtual coordinate with screen position

```
var leap_main = function (frame) {
    gesture_handler(frame);  // function for handling takeoff, landing and yaw
    hand_handler(frame);     // function for handling all other actions [roll,throttle,pitch]
}
```

Figure 9. Leap motion main function on Node.js environment

Current action (type string)		action	DJI function call	Description
index (type int)	action speed (type float)			
0	0	"no_action"	mFlightController.sendVirtualStickFlightController(null)	Drone is either hovering or is on the ground
1	0	"takeoff"	mFlightController.startTakeoff(Callbacks)	Start drone takeoff procedure
2	0	"land"	mFlightController.startLanding(Callbacks)	Start drone landing procedure
3	PalmPosX	"roll"	mFlightController.sendVirtualStickFlightController(FlightController)	Move the drone leftward or rightward
4	PalmPosY	"throttle"	mFlightController.sendVirtualStickFlightController(FlightController)	Move the drone upward or downward
5	PalmPosZ * -1	"pitch"	mFlightController.sendVirtualStickFlightController(FlightController)	Move the drone forward or backward
6	0.5	"yaw"	mFlightController.sendVirtualStickFlightController(FlightController)	Rotate drone clockwise or counterclockwise

FlightController – roll, pitch, throttle, yaw
PalmPosX/Y/Z – the value of palm position on x, y, or z-axis, with z-axis multiplied by -1 to invert the axis

Figure 10. Action map table

The first function inside the main function shown on Figure 9 is for "takeoff", "landing", and "yaw" (rotate drone clockwise and counter clockwise). Two out of four pre-defined gesture provided in the Leap motion API was used by the function. The two pre-defined gesture is circle gesture and key tap gesture as shown on Figure 12 and 13 respectively.

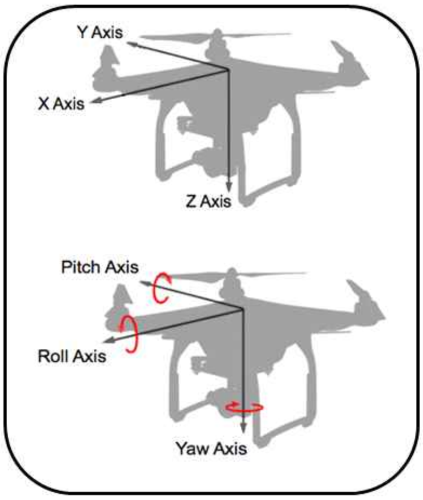

Figure 11. Drone coordinate system

Figure 12. Pre-defined circle gesture for yaw movement

Figure 13. Pre-defined key tap gesture for takeoff and landing

Second function from main function in Figure 9 is for "pitch" (move forward and backward), "roll" (move left and right), "throttle" (move up and down). When a user's hand data is present the function will get the palm position of the hand. In order to increase the accuracy of the system, interaction box provided by Leap Motion API as shown on Figure 14 will be used. The interaction box will normalize the user's palm position in the 3D virtual space. This will be resulted in a palm position ranging from -1 to 1 on the x, y, and z-axis of the 3D virtual space. A threshold on x, y, and z-axis is needed to categorized the gesture that will be given a meaning based on the action map table as Figure 10. Furthermore, the range of values of palm position on the x, y, and z-axis will mean some drone movement, and the value of the palm position will be the movement speed of the drone as shown on Figure 15.

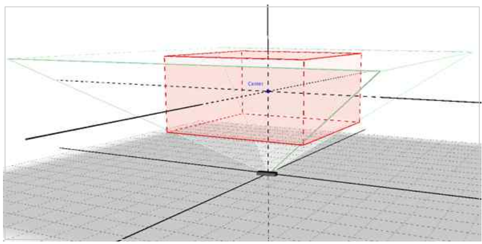

Figure 14. Leap Motion interaction box

axis	threshold (min: -1.00, max: 1.00)	action string	description
x-axis	- 0.40 ~ - 0.95	"roll"	roll to ***left*** or move drone ***leftwards***
	0.40 ~ 0.95		roll to ***right*** or move drone ***rightwards***
y-axis	- 0.60 ~ - 1.00	"throttle"	throttle ***down*** or move drone ***downwards***
	0.60 ~ 1.00		throttle ***up*** or move drone ***upwards***
z-axis	- 0.35 ~ - 0.95	"pitch"	pitch to ***back*** or move drone ***backwards***
	0.35 ~ 0.95		pitch to ***front*** or move drone ***forwards***

Figure 15. Action threshold table

Next implementation is between notebook and the Android device. The implementation uses BLE notification features. Based on the previous implementation method, current action string ("action,action_speed") as in action map table on Figure 10 will be send to Android device through BLE. Before sending or receiving any data, notebook and android device must be connected and this can be done by following the protocol of BLE. If a web-socket uses data packets, BLE protocol uses General attribute (Gatt) as on Figure 16. In order for android application to run the BLE notification feature, a Gatt client must first scan a BLE device and establish a connection to the device similar to the conventional Bluetooth protocol but instead the Bluetooth pairing feature is not compulsory. After a successful connection, the Gatt client will try to discover specific Gatt service that matched the universal unique identifier (UUID) defined by developer. Than it will go through the Gatt service to find the correct Gatt characteristic that also matched the UUID defined by developer. If the correct Gatt characteristic was found the BLE manager will run the BLE notification feature. This all is done on the android device as code on Figure 17.

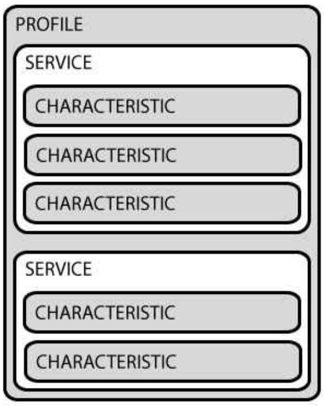

Figure 16. BLE's General attribut (Gatt)

```
List<BluetoothGattService> gattServices = gatt.getServices();
BluetoothGattService service = null;
for (BluetoothGattService gattService : gattServices) {
    Log.i(TAG,  msg: "Service UUID Found: " + gattService.getUuid().toString());
    if (gattService.getUuid().toString().equals(SERVICE_UUID.toString())){
        service = gattService;
    }
}
if (service != null) {
    BluetoothGattCharacteristic characteristic = service.getCharacteristic(CHARACTERISTIC_COUNTER_UUID);
    if (characteristic != null) {
        setCharacteristicNotification(characteristic,  enabled: true);
    }
}
```

Figure 17. BLE code to trigger BLE notification feature

Meanwhile on the notebook, if a BLE notification from android application was triggered, Bleno.js, a node.js BLE module will send the data of current action string ("action,action_speed") for each 60 milliseconds interval with 'utf-8' format as show on Figure 18. The data received on the android application as Figure 19 can be further manipulated in order to control the drone.

```
// send data of current_action to subscribed android device using BLE_notification
LeapDroneBLECharacteristic.prototype.onSubscribe = function(maxValueSize, updateValueCallback) {
    console.log("BLE: device subscribed");
    console.log("BLE: notifying");
    this.intervalId = setInterval(function() {
        updateValueCallback(new Buffer(current_action,'utf8'));
    }, 60); // time is in millisec
};
```

Figure 18. BLE code to send data to android application through BLE notification feature

Figure 19. Printed action as string, for example "takeoff" or "land"

The last implementation needed is between the Android device and the drone. Again by referring the action map table as show on Figure 10. The data of current action string received from BLE notification can be mapped using simple if else or switch function to the drone task (the function calls support by DJI application program interface). By doing this it will tell the drone about what to do relative to the current action (user's hand gesture), and will complete the system implementation.

나) 결과분석

The result of this study can be evaluated by testing the safety and reliability between all components, simplicity of gesture and also by calculating the latency of the system. The main safety of the system is to make sure that the Android application do not crash while the drone is on the air. For the case if Android application crashes, the drone must continue to hover on the air until connection with the drone is establish again. The reliability of the system is solely the connection throughout all the components. Connection between all components must be maintained from the time user starts the application until the user decides to exit the application.

Simplicity of gesture is compulsory for the study to achieve its goal. As a result, the requirement of the system is that, a user must only use one hand to make hand gestures, otherwise the code will skip the function that manipulates user hand gesture data to be used for controlling the drone. Takeoff and landing uses the same gesture as it will be as easy as for a person to turn a switch on and off. Furthermore, making a circle gesture will rotate the drone, and to make it more convenient clockwise circle gesture will rotate the drone clockwise and counter-clockwise circle gesture will then rotate the drone counter-clockwise. Same goes for the pitch, roll, and throttle movements, gesture in Leap motion 3D coordinate system will be matched accordingly prior to the drone 3D coordinate system so that it will be more relevant for users.

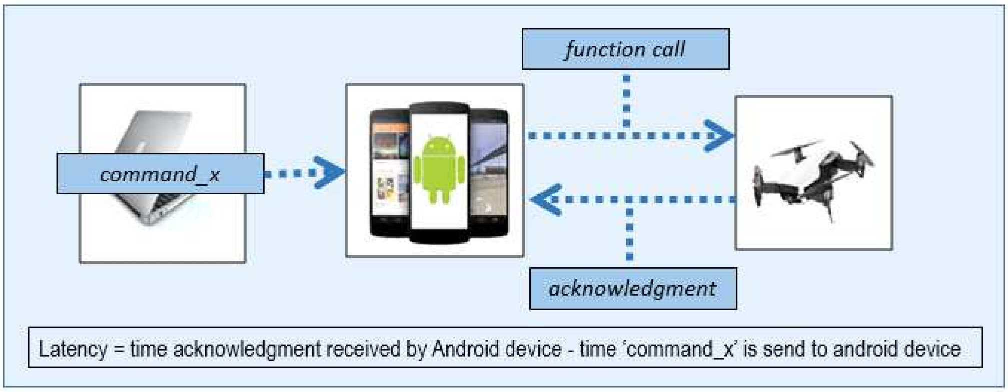

Figure 20. Latency calculation formula

Latency in this study is the time difference between the time for 'command_x' to be send through the Bluetooth Low Energy and the time for a successful function call that is use to move the drone as Figure 20. When calculating the latency, it is also important to consider the distance between all components. Because the Leap Motion device uses USB to connect to the notebook (Mac Book pro), the latency for this two component will be so little, and hard to calculate. Thus, it will not be considered during the latency calculation. For the notebook and Android device, the RSSI can be used to show the range between the two component but because they will only be right next to each other, it will also be useless to take it into consideration. Thus the only thing that was taken into consideration for distance measurement is the horizontal distance between Android device and drone. Horizontal distance between both of this two component was used because it is more accurate compare to the vertical distance theoretically. The latency calculation of the system can be refer as Figure 21. By referring

Figure 21, the average latency that was calculated is 503 millisecond or 0.5 second. It is also been found that the latency of "takeoff" and "landing" are the highest because this two action must be fully completed before the drone can send acknowledgement message to the Android application.

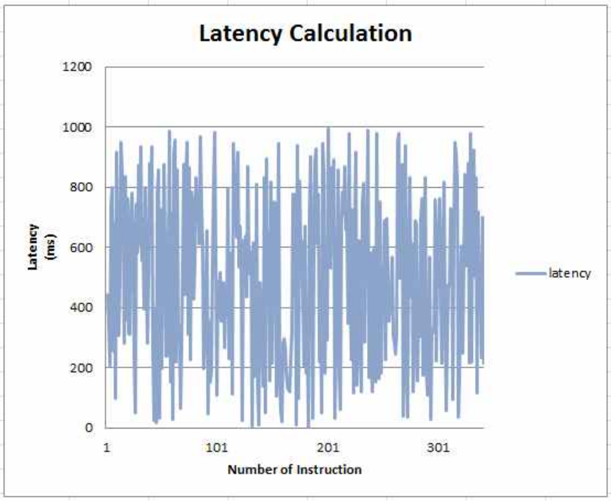

Figure 21. the result of latency calculation

The scenario for the system are as stated below:

1. Run Node.js code on the notebook to enable both Leap Motion features and BLE features.

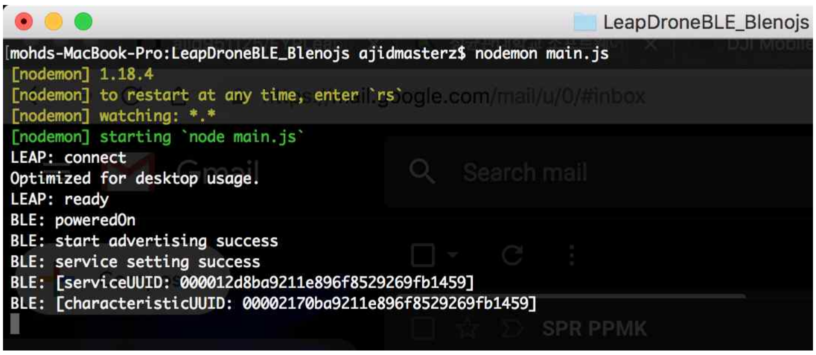

Figure 22. BlenoJS run on terminal

2. On the Android device, connect to DJI Mavic Air via Wi-Fi.
3. Open the android application

4. If the DJI Mavic Air is present, button will be enable. Press the button when it is enable.

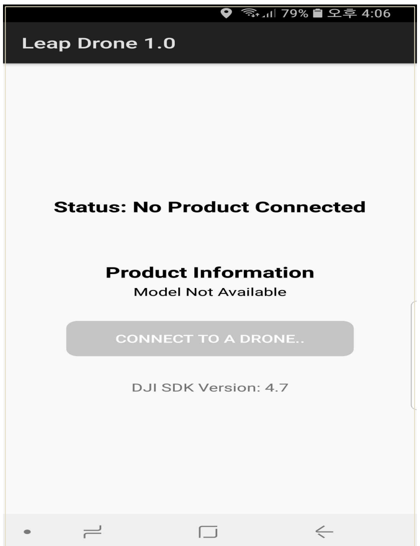

Figure 23. Android application main view

5. Search for BLE devices, and select a particular device named by developer, ('Leap Motion')

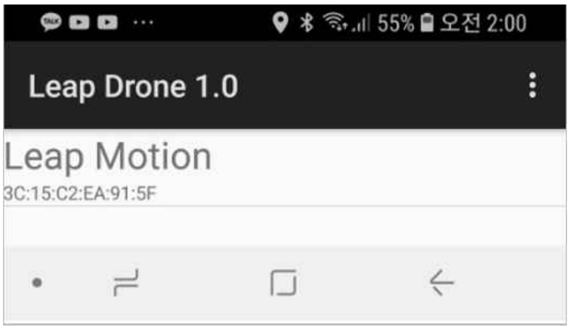

Figure 24. A view layout to connect to BlenoJS

6. A successful connection with the notebook will start the BLE notification and started the FPV activity

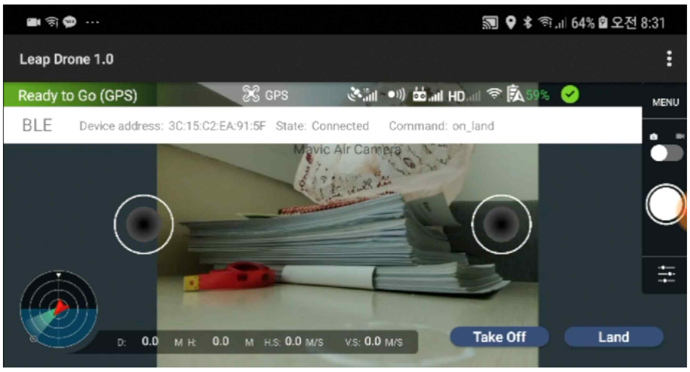

Figure 25. FPV view on Android application

7. Lastly, the drone can be control, and here are an example of the system being tested outdoor using all the implementation that was discussed.

Figure 26. Drone control in real world

5. 결론 및 소감

가) 결론

As the study main purpose is to create a system that uses simpler hand gesture and making complicated drone control look easier, further implementation will be needed to increase the flexibility by letting user configure and record any hand gesture suiting to their personal taste. This makes it easier for amateur drone lover to avoid complexity of a typical physical remote controller stick. In addition, it will be able to provide a healthier environment for drone markets as users whom is handicapped person also would be able to control a drone in the future. Also the future development must focus on widening the drone control distance, so that the system also can be used for un-ordinary drone missions such as search and rescue which needed wider drone control range.

The things that was not meeting the expectation about this study is on developing and testing the system with more features that also will be resulted in the need of more gesture implementation. Extra features are such as camera’s gimbals positioning, camera capture and video recording, unorthodox drone moves such backflip or moving drone with 360 angle surrounding the user, and etc. It was also not easy to develop a system involving a drone because of flight restrictions and the surroundings in South Korea.

나) 소감

The study teaches about some basics of network and IOT development, and the way of implementation which differ between components. It was also a fun experience to be able to develop a system that involve a drone and leap motion. The crucial thing is that to understand the 3D coordinate system which is the core part around the application programming interfaces (API) for both components. By understanding the 3D coordinate system it will be easier to develop more feature that can be executed by the specific gesture defined by developer. Overall, it was a good experience to develop a system that involve multiple components, and also positive to improve a person programming skills based on network and IOT system development.

6. 참고문헌

[1] 무인항공기 위키백과 https://ko.wikipedia.org/wiki/무인_항공기
[2] 드론 나무위키 https://namu.wiki/w/드론
[3] Leap Motion Wikipedia. https://en.wikipedia.org/wiki/Leap_Motion
[4] https://www.dji.com/mavic-air/info#specs
[5] JinHa Hwang, Jeong Il Yang, TaeSeok Kim, and ChoongSeon Hong, "Easy Drone Control System using Motion Recognition", Dept of Computer Engineering, Kyung-Hee University, 2017.
[6] Zhenfei Zhao, Hao Luo, Guang-Hua Song, Zhou Chen, Zhe-Ming Lu, and Xiaofeng Wu, "Web-based interactive drone control using hand gesture", American Institute of Physics, 2018.
[7] Ayanava Sarkar, Ketul Arvindbhai Patel, Ganesh Ram R.K., and Geet Krishna Capoor, "Gesture Control of Drone Using a Motion Controller", https://www.researchgate.net/publication/301800528_Gesture_control_of_drone_using_a_motion_controller, 2016.
[8] Hojin Ha, Sangjae Park, and Seungkeun Lee, "Design of Mobile Application for Virtual Reality-based Tour Information Using Leap-Motion and Unity 3D", Department of Information, Communications and Broadcasting Engineering, Halla University, 2017.
[9] Ram´on A. Su´arez Fern´andez, Jose Luis Sanchez-Lopez, Carlos Sampedro, Hriday Bavle, Martin Molina, and Pascual Campoy, "Natural User Interfaces for Human-Drone Multi-Modal Interaction", International Conference on Unmanned Aircraft Systems (ICUAS), 2016.
[10] Martin Catherine, "Motion in the Field: A Study of Movement in Computer Science", Syracuse University Honors Program Capstone Projects. 904, 2015.