GCS Interface Mission Load Reduction Design through Pilot Gaze Point Analysis

Sedam Lee and Yongjin Kwon*
Ajou University, Department of Industrial Engineering, Suwon, South Korea
Email: lsd2547@ajou.ac.kr, yk73@ajou.ac.kr

Jinyeong Heo and Jooyoung Kim
LIG Nex1 Co., Intelligent/Autonomous Control SW Team, Seongnam, South Korea
Email: jinyeong.heo@lignex1.com, jooyoung.kim@lignex1.com

Abstract—**Due to recent technological advancement, the interest in the commercialization of unmanned aerial vehicles(UAVs) is increasing rapidly. UAVs are continuously being studied for various applications and purposes. As the purpose and use of UAVs expand, the demand for the pilot ability for safe and successful flight is also increasing. Due to the nature of the UAVs, it takes a long time for the pilots to become proficient. Despite, if a difficult situation arises during a flight and the pilot's workload unexpectedly rises, an accident may occur. This is mainly due to mission load hike and inexperienced operation. Therefore, there is a need for a new GCS interface design to reduce the mission load, while increasing the mission success rate. In this paper, we propose a new method to design a more efficient GCS interface, and the new design has bene verified for its efficiency. The pilot gaze was tracked during mission operation using the newly designed GCS interface. The interface components were analyzed through the tracked gaze, and a GCS interface was evaluated. As a result of deriving the arrangement method with the shortest gaze path, the new GCS interface is found to be an efficient arrangement. At the same time, the new interface design shares the same ergonomic design principles, which proves to be fulfilling the design purposes.**

Index Terms—**GCS interface, mission load, pilot gave, UAV flight, fatigue reduction, operation efficiency**

I. Introduction

Unmanned Aerial Vehicles (UAVs) have been continuously researched and developed since the dawn of 4th industrial revolution. As the commercial use of UAVs is increasing due to the increase in usability along with technological development, it is expected that they will be used for many diverse purposes in the future.

UAVs, which are being used in various fields like this, are different from existing manned aircraft in which a pilot boards an airplane. The UAVs have a separate GCS (ground control system) to allow the operator to check the operation information of the unmanned aerial vehicle on the ground. In addition, when a command is issued through the GCS, the unmanned aerial vehicle performs a mission according to the command. The most direct way to input commands to UAVs from GCS and to visually check operation information is through GUI (graphical user interface)'. The GUI provides functions such as input and output in an easy-to-understand form for users. In particular, GUI refers to an interface that is visually provided through graphic elements using shapes and colors on the display [1].

GCS interface differs in information and operation method depending on the purpose of use and development company. The interface that provides the camera screen is important for the GCS screen of the unmanned aerial vehicle for the purpose of video shooting. In addition, in the case of UAVs used for facility and forest management, the function to control the aircraft is most important [2]. Therefore, the GCS interface components provide functions and information suitable for the purpose. In the case of UAV operation using a new GCS interface rather than the existing GCS interface, even pilots with a lot of experience in UAV operation may experience a mission load due to the unfamiliar GCS interface, and accidents may occur due to operation errors [3] [4].

Therefore, the purpose of this study is to analyze the existing GCS interfaces with different purpose of use and development environment, and design a new interface so that the pilot can reduce the mission load that occurs during UAV operation. Also, it is expected that the pilots can utilize the interface more efficiently [5][6][7]. An experiment was conducted to design a GCS interface that can minimize UAV operation negligence and control errors and make quick and clear situational judgment. Through this, an efficient arrangement design for the interface was derived. The structure of this paper is as follows. In the case of Chapter 2, when controlling the unmanned aerial vehicle using the existing GCS interface, the pilot gaze is tracked and utilized, and the high-priority interface is analyzed. Through this process, the necessary requirements are derived for the new GCS interface. In Chapter 3, the basic GCS interface that satisfies the derived requirements is materialized. In Chapter 4, the experiment to identify a more efficient interface is discussed, which is according to the arrangement of

Manuscript received April 14, 2022; revised August 1, 2022.
*Corresponding author" Yongjin Kwon

doi: 10.18178/ijmerr.11.9.710-717

interface components. Finally, Chapter 5 describes the results and conclusions of this paper.

II. Derivation of Requirements through Eye Tracking

A. Establishing an Environment for Eye Tracking

To design an efficient GCS interface, 'Mission Planner', the most popular GCS interface among commercialized GCS interfaces, was used. An experiment was conducted to identify the interface components that are most frequently recognized and utilized by the pilots. A simulation environment was implemented to track the locations that the pilots stare when performing the actual mission. Through this, the interface components that the pilots stare at during the mission operation can be identified. When operating a mission using an actual UAV, a simulation is used as shown in Fig. 1 to prevent disturbances caused by various emergency situations that are difficult to cope with.

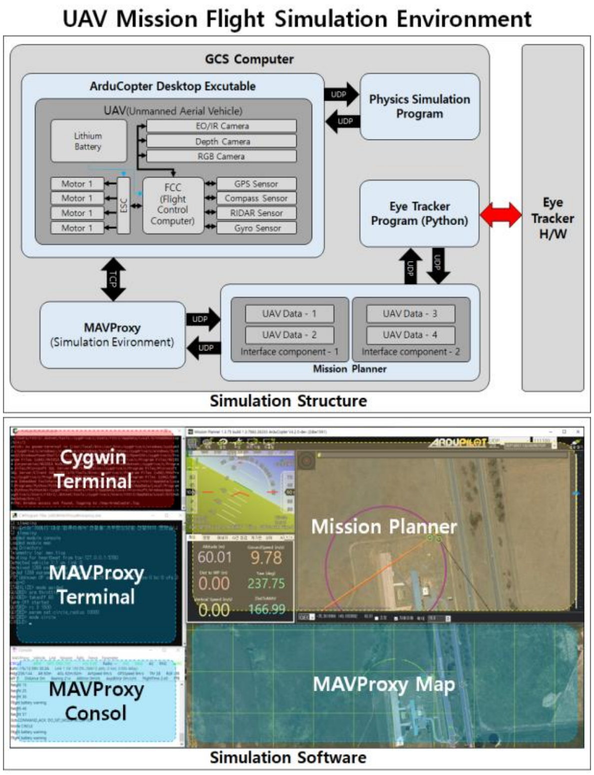

Figure 1. Experimental environment implementation.

The simulation program and GCS Interface, Mission Planner, were linked together. Afterwards, it was designed to extract pilot gaze tracking data from the Eye Tracker as a CSV file, using a Python language. In order to track the pilot gaze and identify the interface components that are being stared, each component constituting the interface is divided into ROI (region of interest). Mission Planner's interface screen is provided in two ways: a flight plan screen and a flight operation screen. Two screens were divided into ROIs as shown in Fig. 2.

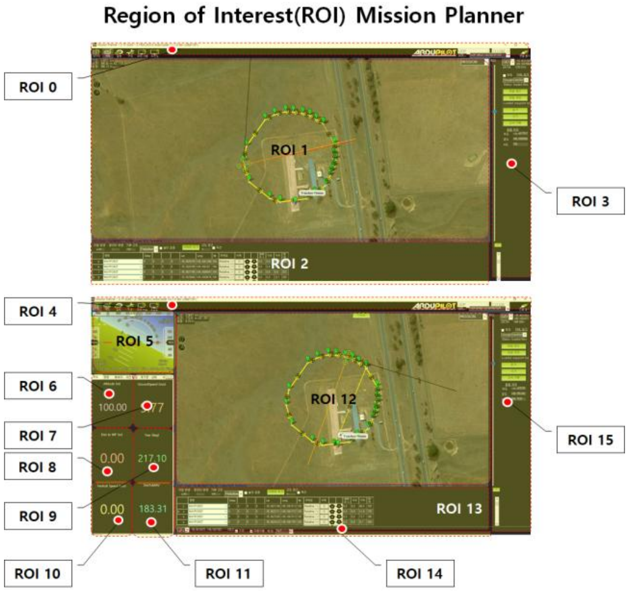

Figure 2. ROI classification in mission planner.

Table I shows the interface components for each ROI. ROI_5 is a grouping of similar or related interface elements. On the contrary, there is also an interface element that provides only single information.

TABLE I. Interface Components Belonging to the ROI

Display	List	Interface
Flight Plane Display	ROI_0	GCS Option
	ROI_1	Mission Plane Map
	ROI_2	Waypoint Setting
	ROI_3	Waypoint Option
Flight Operation Display	ROI_4	GCS Option
	ROI_5	Primary Flight Display
	ROI_6	Altitude
	ROI_7	Ground Speed
	ROI_8	Distance To WP
	ROI_9	Yaw
	ROI_10	Vertical Speed
	ROI_11	Distance To Mav
	ROI_12	Operation Map
	ROI_13	Waypoint Information
	ROI_14	GPS Information
	ROI_15	Waypoint Option

B. Pilot Eye Tracking

The simulation for tracking pilot gaze was conducted for about 15 minutes from mission planning to mission operation. The pilot gaze point was extracted at regular intervals. As for the gaze point, as shown in Fig. 3 and Fig. 4, the gaze points on the mission plan screen and the mission operation screen were visualized on the Mission Planner screen.

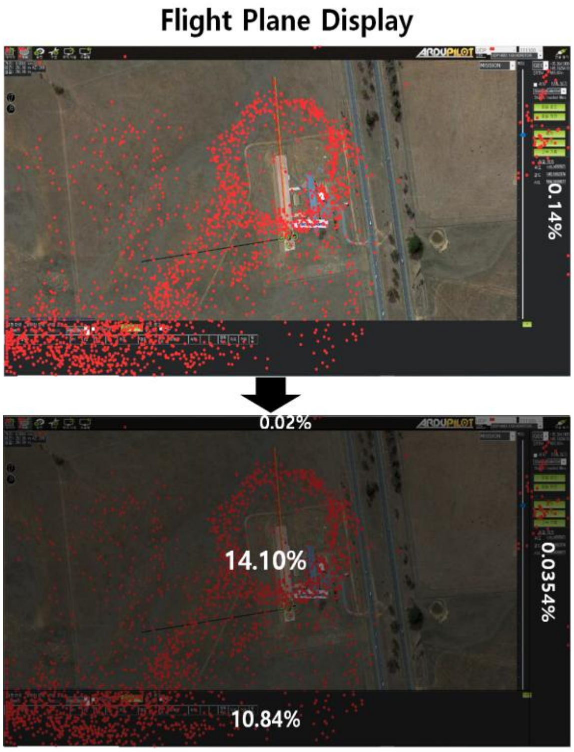

Figure 3. Visualization of gaze point on the mission plan screen.

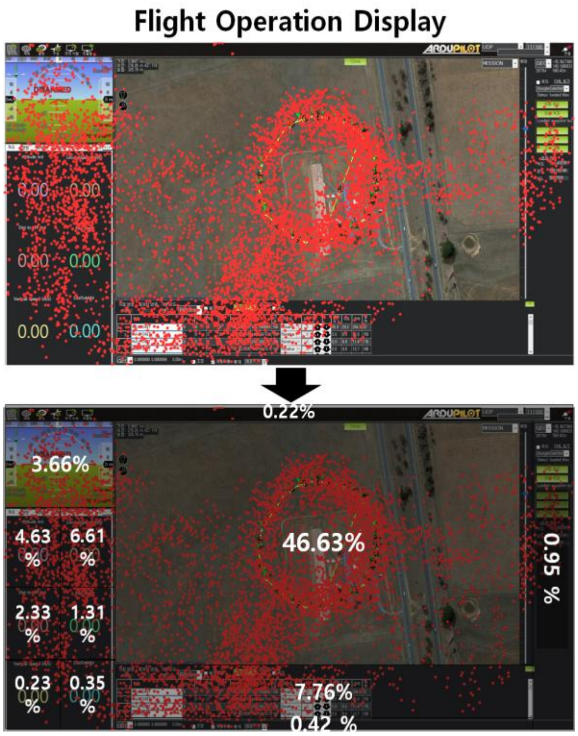

Figure 4. Visualization of gaze point on the mission operation screen.

The gaze points were expressed in the form of a percentage value, and the data were collected for 15 minutes. Fig. 5 is a graph showing the degree of gaze point of the pilot by ROI.

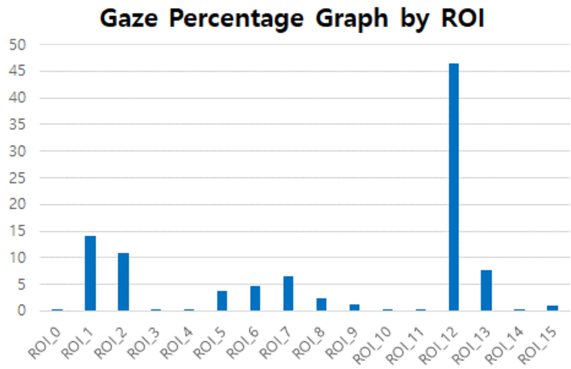

Figure 5. Gaze percentage graph by ROI.

Through this, the important interface components including functions and information, which need to be considered in designing the GCS, were derived. These were used as essential requirements. In addition, user requirements were identified through pilot interviews who have expertise in controlling UAVs. Table II shows the GCS interface components derived from essential requirements for interface design and user requirements. It is necessary to verify whether the requirements derived after interface design can be satisfactory.

TABLE II. INTERFACE DESIGN REQUIREMENTS

Number	Interface Type List
1	Mission Map
2	Primary Flight Display
3	Direction Indicator
4	Waypoint Command
5	Camera Screen
6	Attitude Information
7	GPS Information
8	Ground Speed
9	Air Speed
10	Altitude
11	Vertical Speed
12	Battery Information
13	Available Time
14	Voltage Information
15	Motor Power Information
16	Waypoint Information
17	Behavior command
18	Joystick

III. PLACEMENT EFFICIENCY OF INTERFACE COMPONENTS

A. *Basic GCS Interface Design*

In order to design the GCS interface that satisfies the requirements derived from the pilot gaze point analysis, the following procedure was carried out. Interface components that provide similar or related information and functions are grouped. Table III classifies the interface components into the Task Breakdown Structure (TBS) so that the interface components for each group can be designed together.

TABLE III. TBS GROUPING OF INTERFACE COMPONENTS

Level 1	Level 2	Level 3
GCS Interface	Main Display	PFD
		Direction Indicator
		Way Point Command
		UI Button
		Camera Screen
		Notice
		Mission Map
	Flight Panel	Flight Information
		Motor Power
		Way Point Setting
		Basic Information
	Convenience Panel	Behavior Command
		Joystick

Among the TBS, the Main Display, Flight Panel, and Convenience Panel belonging to Level 2 are the top level components of interface design. In the case of Level 3, it was classified into components that provide detailed functions and information. The interface was designed based on the interface components classified by the TBS. The interface type is generally defined as the interface with the structure and location that is most easily accessible. This is defined as the Basic GCS Interface, as illustrated in Fig. 6.

Figure 6. Basic GCS interface design.

In order to check whether the designed Basic GCS interface satisfies the above-mentioned requirements, a list of requirements was used as shown in Table III. Through the requirements list, it is indicated whether the interface design requirements are reflected in the Basic GCS interface.

TABLE IV. CHECK INTERFACE DESIGN REQUIREMENTS

Requirements List	included group	Satisfy
Mission Map	Main Display	O
Primary Flight Display	Main Display	O
Direction Indicator	Main Display	O
Waypoint Command	Main Display	O
Camera Screen	Main Display	O
Attitude Information	Flight Panel	O
GPS Information	Flight Panel	O
Ground Speed	Flight Panel	O
Air Speed	Flight Panel	O
Altitude	Flight Panel	O
Vertical Speed	Flight Panel	O
Battery Information	Flight Panel	O
Available Time	Flight Panel	O
Voltage Information	Flight Panel	O
Motor Power Information	Flight Panel	O
Waypoint Information	Flight Panel	O
Behavior command	Convenience Panel	O
Joystick	Convenience Panel	O

B. Interface Arrangement Plan According to Location

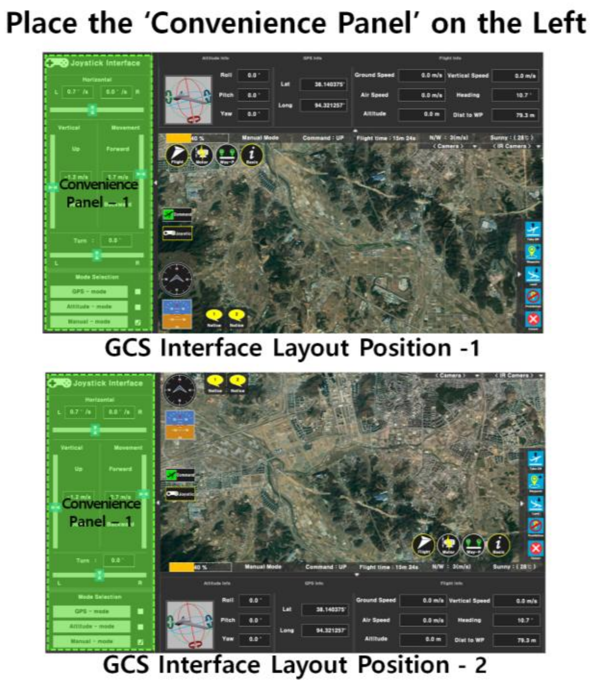

Figure 7. Interface component arrangement plan _1.

This section shows the utilization of quantitative and qualitative data for efficient interface design. For quantitative data, the pilot gaze path was used. When operating a mission, the interface layout plan with the pilot's shortest line of sight is selected. In order not to damage the basic components of the interface, the Flight Panel is arranged based on the Convenience Panel as shown in Fig. 7 and Fig. 8. Here, four arrangement methods are presented as shown in the figures.

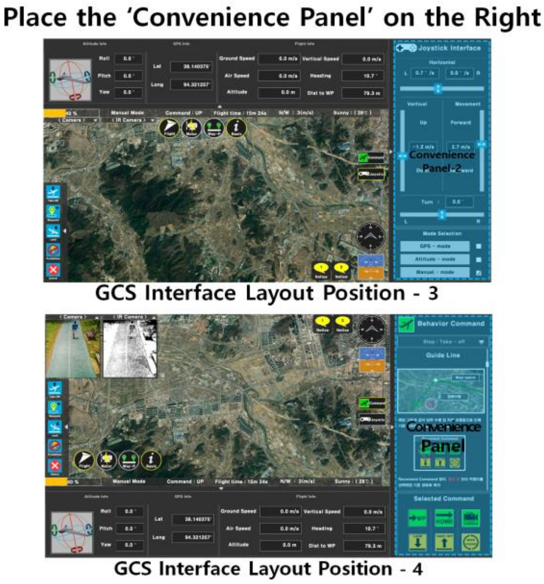

Figure 8. Interface component arrangement plan_ 2.

When performing the same mission, the most efficient design is the interface layout with the pilot's shortest line of sight. This is because the pilot moves his gaze to find information and functions on the screen while operating the UAV. In this case, the information necessary for mission operation should be most conspicuous, and the more efficient it is, the less likely it is that the pilot will glance the display screen. In fact, as a result of the experiment, the time required for the eyes to stay on the screen is reduced unnecessarily to find interface information. Therefore, in this study, verification through simulation was conducted to find the most efficient interface arrangement method that does not damage the component design of the Basic GCS Interface.

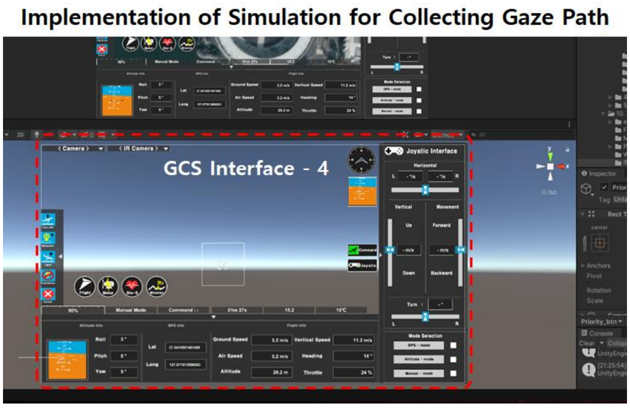

Figure 9. Experimental environment for deriving an efficient interface arrangement plan.

In order to analyze the pilot gaze path during actual UAV operations, a simulation environment was implemented as shown in Fig. 9. The simulation was implemented using the Unity Program. The UAV model for the simulation utilized a UAV model that includes a dynamic model, sensors, and equipment, and was designed to visualize various sensor values from the UAV model on the screen. In order to derive the interface arrangement method with the shortest line of sight among the four interface arrangements, it is necessary to extract the line of sight of the pilot under the same conditions. Therefore, the route plan of the unmanned aerial vehicle was unified in the simulation, and all were set to perform the same mission. By collecting only the gaze path from the start to the completion of the mission operation, the gaze point for each of the four arrangements can be extracted as a CSV file. The gaze point has the form of an image coordinate (x, y) value in every predetermined time. Fig. 10 shows the mission planning for gaze path extraction.

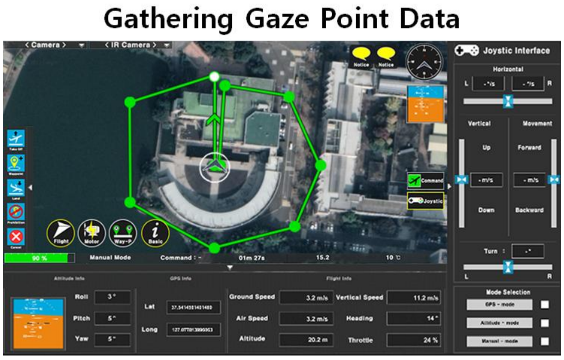

Figure 10. UAV mission planning for gaze path extraction.

IV. Efficient Interface Layout Derivation

A. Gaze Point Extraction

Through the extracted gaze point data (i.e., a CSV file), the actual pilot gaze on the GCS interface screen was visualized. Visualization is indicated by a red dot on the image coordinates. The CSV file is extracted as image coordinates (x, y) for every frame as shown in Fig. 11.

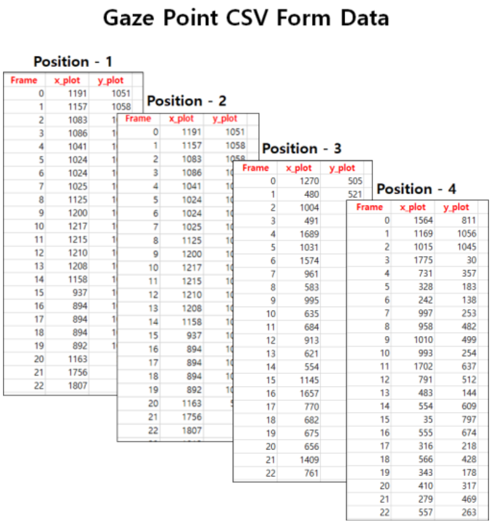

Figure 11. Gaze Point Image Coordinates CSV Data.

Fig. 12 shows the viewing point visualization of the screen, corresponding to 'Position_4' among the four interface arrangements.

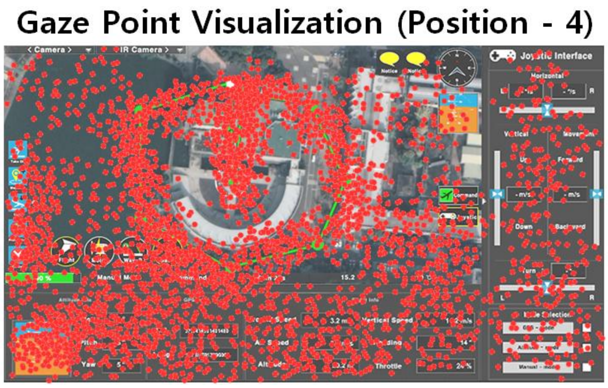

Figure 12. Position 4's gaze point visualization.

When the pilot gaze point is extracted, noises may occur due to trembling of the gaze or closing of the eyes. Also, even if you are gazing the same points, the gaze point is extracted not at the exact same point but around it due to tremor of the pupil. It is necessary to determine the gaze path through the corrections. For this purpose, the gaze points having the image coordinate values extracted within a certain range according to the gazed time sequence were clustered.

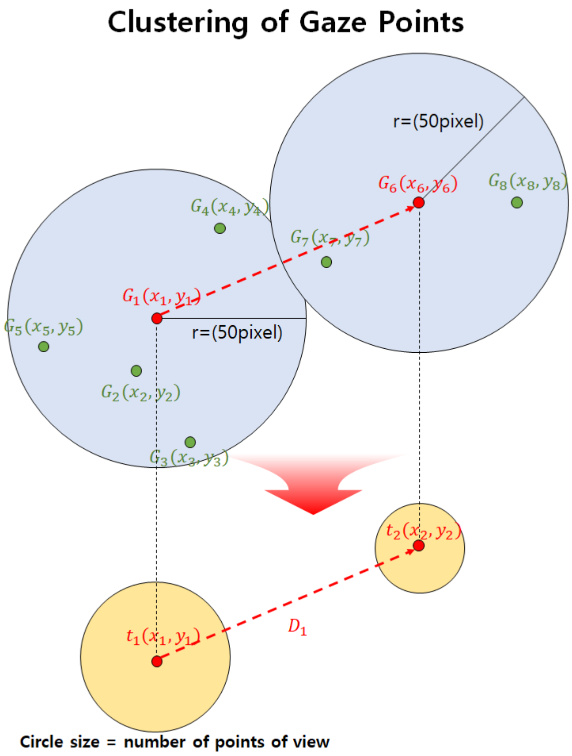

Figure 13. Clustering of Gaze Points.

The clustered gaze points were classified as shown in Fig. 13, and the size of the cluster gaze points indicates the number of clustered gaze points. Through this, the visualization screen of Position_4 clustered appears as shown in Fig. 14.

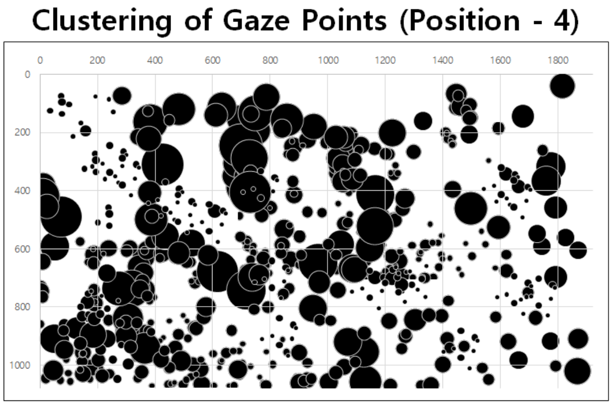

Figure 14. Clustering visualization of Position_4.

In order to grasp the gaze path of the clustered gaze points, the distance between the gaze points was analyzed. In order to derive the distance between the points, it was defined as follows. When the order of the cluster of the first gaze point is t_1, the central coordinate of the cluster gaze point of t_1 is defined as $t_1(x_1, y_1)$,and the cluster center coordinate of the second order, t_2 ,is defined as $t_2(x_2, y_2)$. In this case, the distance from $t_1(x_1, y_1)$ to $t_2(x_2, y_2)$ is defined as D_1. This is illustrated in Fig. 15.

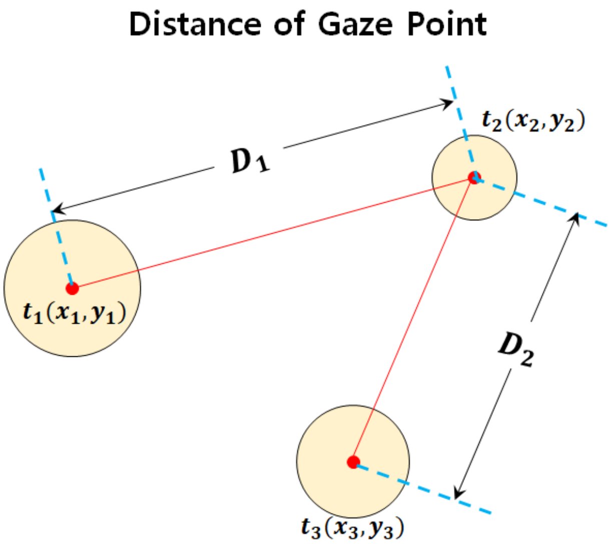

Figure 15. Measure the distance between gaze points.

Assuming that the order of the last cluster gaze point is t_{n+1}, the distance from $t_n(x_n, y_n)$ to $t_{n+1}(x_{n+1}, y_{n+1})$ is D_n. In this case, the method to obtain D_n is the same as in Equation 1.

$$D_n = \sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2} \qquad (1)$$

The most efficient interface arrangement method among the four is more effective as the pilot checks the interface and the short line of sight through which the next interface is identified. However, even if the length of the entire gaze path is the shortest, the average gaze path is long because the number of clustered gaze points is small. Then, the corresponding interface arrangement method is not a relatively efficient method. Therefore, it can be said that the most efficient arrangement is with relatively short two line-of-sight paths by analyzing the pilot's entire line of sight path and the average line of sight path. The total gaze path according to the interface arrangement method was defined as T(D). The average gaze path was defined as E(D), and the number of cluster gaze points was defined as (n).

$$\mathrm{T(D)} = \sum_{i=1}^{n} D_i \quad (2)$$

$$\mathrm{E(D)} = \frac{T(D)}{n} \quad (3)$$

Table V shows the total gaze path, average gaze path, and the number of cluster gaze points for the four interface arrangement methods.

TABLE V. GAZE PATH ANALYSIS

Interface Position	Cluster Point (n)	T(D)pixel	E(D)pixel
Position 1	718	460984.72	642.04
Position 2	683	454657.93	665.47
Position 3	673	472109.50	701.50
Position 4	724	457312.93	631.37

In the case of Position_2, the length of the entire gaze path, which is **T**(**D**), is the shortest. However, since the number of clustered gaze points is small, the average gaze path is longer than Position_4. In the case of all gaze points, the number of gaze points is the same because the gaze points were collected for each frame of the same eye while operating the same task. Therefore, the small number of cluster points simply means that the number of gaze points decreased during the clustering process because many gaze points were extracted at similar locations.

B. Gaze Point Analysis

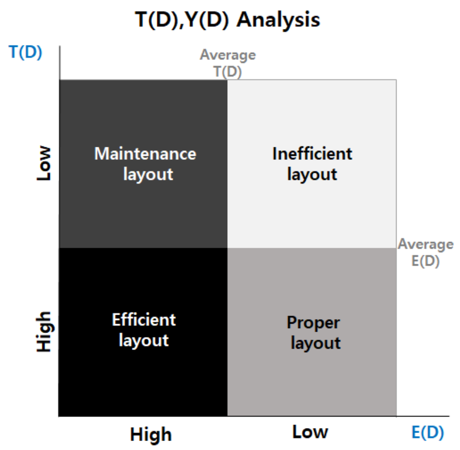

Figure 16. Interface analysis using IPA technique.

Among the four interface arrangement, there are two indicators that can be used to derive an efficient interface. In order to derive the most efficient interface arrangement method, the IPA (important performance analysis) method was used, as in Fig. 16 [8]. An efficient interface was analyzed by indicating each item as quadrants on a plane coordinate with the values of T(D) and E(D) as the X and Y axes [9][10].

Fig. 17 shows when **T**(**D**)and E(**D**) from Position_1 to Position_4 are applied to the quadrant. That is, it is determined that Position_4 located in 'Efficient Layout' is the most efficient. On the other hand, the interface arrangement method of Position_3 is judged to be the most inefficient.

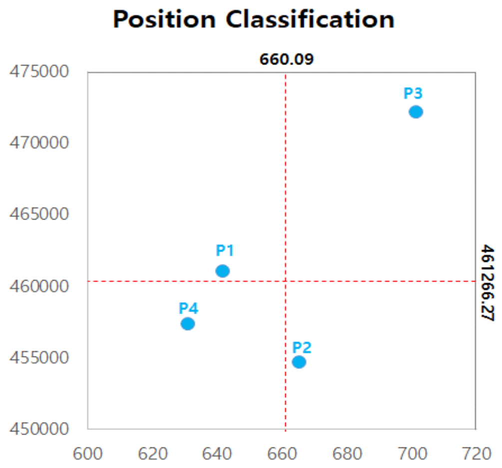

Figure 17. Efficiency analysis from Position_1 to Position_4.

V. RESULTS AND CONCLUSIONS

In this study, an experiment was conducted on the GCS interface design method to provide the most efficient interface to the pilot when operating an unmanned aerial vehicle. After deriving the GCS interface requirements based on the gaze point, which is the quantitative data extracted by tracking the gaze of the actual pilot, the Basic GCS Interface was designed. Afterwards, according to the arrangement method of the interface components, four arrangement methods were selected to identify more efficient interfaces. The gaze path for each was analyzed. In order to derive the most efficient arrangement method among the four arrangements of the GCS interface, a simulation environment was implemented and the pilot gaze path was traced. The most efficient interface arrangement method was derived through the experiment. It was confirmed that the arrangement method that provides an efficient interface through the optimal gaze path was the GCS interface of Position_4. The experiments found that the existing layout method of an interface familiar to the pilot was more effective than the unfamiliar layout method. In the end, it was confirmed that the interface designed by considering the user's experience is the most important item in interface design. Also, the layout method derived through the gaze path has a structure very similar to the GCS interface design based on the existing ergonomic theory. These experiments proved that the familiar form of GCS was effective through quantitative experimental results. It means that most of the existing

GCS interfaces are providing an efficient form. The GCS interface designed later can be used to design the GCS interface for controlling not only UAV, but also various unmanned objects such as unmanned surface vehicles and unmanned ground robots. It is expected to be utilized in the development of an efficient interface that can reduce operator fatigue, task load, and prevent accidents.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Authors conducted the research; analyzed the data; and wrote the paper. All authors had approved the final version.

Acknowledgment

This research was supported by Unmanned Vehicles Core Technology Research and Development Program through the National Research Foundation of Korea (NRF) and Unmanned Vehicle Advanced Research Center (UVARC) funded by the Ministry of Science and ICT, the Republic of Korea (Grant Number: 2020M3C1C1A01084900). This research was also supported by a grant from LIG Nex1 Company (Grant Number: LIGNEX1-2021-0621).

References

[1] T. Kim, “A reconfigurable UAV ground control system,” Mater’s Thesis, Korea Aerospace University, 2011.

[2] D. Stone, C. Jarett, M. Woodroffe, and S. Minocha, *User Interface Design and Evaluation*, Morgan Kaufmann Publishing Co., Los Altos, CA, USA, 2005.

[3] J. Huddlestone, D. Harris, D. Richards, S. Scott, and R. Sears, “Dual pilot and single pilot operations-Hierarchical task decomposition analysis of doing more with less,” *International Conference on Engineering Psychology and Cognitive Ergonomics*, vol. 9174, pp. 365-376, 2015.

[4] NATO, STANAG 4586, “*Standard Interface of UAV Control System (UCS) for NATO Interoperability*,” Ed.1, 2004.

[5] M. Zeiller, “A case study based approach to knowledge visualization,” in *Proc. the Ninth International Conference on Information Visualization*, 2005.

[6] E. Seo, "The suggestion for the design of eye tracker to promote the study on the gaze tracking interface," *KIPAD*, no. 50, pp. 145-152, 2017.

[7] J. Lee, “3D user interface using eye-tracking,” Master’s Thesis, Dept. of Computer Science & Engineering, Konkuk University, 2001.

[8] C. Seo, J. Lee, and S. Choi, “A study on improvement of User Interface (UI) based on multi-point replay system,” *Journal of Broadcast Engineering*, vol. 24, no. 2, pp. 341-352, 2019.

[9] NATO, “*STANAG 4586 NAVY (Edition 2) - Standard Interface of UAV Control System (UCS) for NATO Interoperability*,” Ed.2, 2007.

[10] S. Lee and Y. Kwon, “Safe landing of drone using AI-based obstacle avoidance,” *International Journal of Mechanical Engineering and Robotics Research*, vol. 9, no. 11, pp. 1495-1501, 2020.

Sedam Lee has many years of engineering experience in industrial engineering. He has conducted research applied to industrial engineering concepts in military and simulation. He has a major interest in the fields of simulation, unmanned systems, and UAV. He has studied his master's degree at Ajou University.

Jinyeong Heo received hi Ph.D. degree in industrial engineering from Ajou University, South Korea in 2022. He is currently a research engineer at Intelligent/Autonomous Control SW Team of LIG Nex1. His main research interests are mission planning system, object detection and reinforcement learning.

Jooyoung Kim received her M.S. degree in Computer Science from Ewha Womans University, South Korea in 2009. She is currently a chief research engineer at Intelligent/Autonomous Control SW Team of LIG Nex1. Her main research interests are autonomous control, swarm control and AI.

Yongjin Kwon has many years of engineering experience in industrial and academic settings. He has extensive experience & practical knowledge in current design, manufacturing and quality control. His work has been cited a number of times in high profile journals. He is currently a professor in the Department of Industrial Engineering at Ajou University. Prior to joining Ajou, he was on the faculty of Drexel University.