

A New Software Architecture for Drone Flight and Analysis of Accident

MyungHo So¹, Kyong Hoon Kim² and Ki-II Kim^{3*}

¹*Aero Master Corporation, Sacheon, Korea*

²*Department of Informatics, Gyeongsang National University, Jinju, Korea*

³*Department of Computer Science and Engineering, Chungnam National University, Daejeon, Korea*

**kikim@cnu.ac.kr*

Abstract

Compared to typical analysis for aircraft flight, flight analysis of Drone is mostly dependent on the record without any description of pilot. Also, there are not too much information for accident or special events yet. So, it is demanded to display the flight information in realistic view to recognize several situations as well as provide the accident analysis/management tool for them. However, current Drone analyzers provide mentioned functions partially or not at all. To address this problem, we propose a new software architecture to display in three dimensional simulator and suggest analysis tools for the possible cause of accident where drastic changes are identified. These analysis is based on databased which is managed by user input. We present implementation issue and experimental results. Finally, we compare the existing simulator with our new one in mentioned features.

Keywords: *Drone, Simulator, Accident analysis*

1. Introduction

Recently, Drone provides useful information for many applications such as entertainment and surveillance. Due to unmanned operation, all activities need to be recorded to recognize whether the given mission is completed or not. Also, recording can be used to identify the unexpected events such as accident.

But, the most of current software for drone [1-3] provide visualization for mission during flight in a form of graph and data sheet. So, it is not easy to get accurate information for their flight. Also, as compared to aircraft, there is no details information for events. This means that the user cannot get any information about what are happened in Drone. Table 1 illustrates the current analysis tool for Drone. As you can see, three dimensional display and accident analysis are not current available in most models.

To solve this problem, in this paper, we propose new software to display the flight information as well as analyze the accident in appropriate way. In the proposed software, flight information is displayed in three dimensional simulator with terrain information. Also, accident analysis has accomplished through database which is controlled by user input. Table 2 shows the example of abnormal case which can happens during Drone flight. Each item has normal range which is given by maximum and minimum value. By including two functions, we present the software architecture and then implementation issue. Finally, experimental results are shown to prove proper operation.

The paper is organized as follows. In section 2, we describe the software architecture and details for each component. The implementation result and experimental results are explained in section 3. Finally, we make a conclusion.

Received (July 15, 2016), Review Result (October 31, 2016), Accepted (September 14, 2017)

Table 1. Comparison of Analysis Tools

| Manu- facturer | Model | Flight information | Analysis Tools | Functionalities | | | | |
|-------------------|----------|-----------------------|-------------------|------------------|-----------|---------------|-------|----------------------|
| | | | | 3D Simulation | 2D Map | Data Sheet | Graph | Accident Analysis |
| 3DR | 3DR SoLO | X | X | | | | | |
| Parrot | Bebop | O | O | X | O | O | O | X |
| | AR Drone | | | | | | | |
| CHEERSON | CX-10 | X | X | | | | | |
| SYMA | X5C | X | X | | | | | |
| DJI | Pahntom3 | O | O | X | O | O | O | X |
| | Pahntom3 | | | | | | | |
| | Inspire1 | | | | | | | |

2. Related Work

In this section, we briefly summarize the research effort for Drone simulators. First of all, the authors presented the structural and functional requirements of Unmanned Aerial Vehicle (UAV)[4]. Based on this requirement, they describe the training system for UAV. Their UAV simulation system consists of three layer under hierarchy of the real time simulation system. Under this architecture, sequence diagram, console interface, flow chart to control simulator computer and simulation analysis are given.

Second paper describe ANGEL project (Aerial Network Guided Electronic Lookout)[5] which aims at system engineering to the design, development, testing and implementation of a quadrotor unmanned aerial vehicle. The objective is to make use of general system approach to design and implement UAV. Through this approach, different from typical existing scheme, it is feasible for use in real world. The controller of ANGEL project is based on MATLAB and Simulink. The authors presented the test cases to show successful operation of the craft.

In addition to mentioned simulator, similar to ANGEL project with MATLAB and Simulink, the authors presents the composition and function of the flight simulation system corresponding to characteristics of UAV flight simulation in simulation training system[6]. Fight control as well as navigation model is developed with MATLAB and Simulink with low complexity. For the flight control model, mathematical modeling, control law, and navigation model are explained and implemented. But, there is no experimental result and analysis for them.

Table 2. Example of Abnormal Case

| Type | Item | | Value | |
|-----------|-------------|---------------|----------------------------------|-------|
| | | | Min | Max |
| Signal | GPS | | GPS Signal off | |
| | Battery-Low | | Battery Low Warning Signal on | |
| | RC | | Remote Control Warning Signal on | |
| Overvalue | Altitude | | 0ft | 200ft |
| | Airspeed | 0 knots | 15 knots | |
| | Distance | 0 km | 1 km | |
| | Rollrate | ± 0 deg/s | ± 60 deg/s | |
| | Pitch rate | ± 0 deg/s | ± 40 deg/s | |
| | Yaw rate | ± 0 deg/s | ± 180 deg/s | |
| | Sink rate | ± 0 fps | ± 13 fps | |

Furthermore, UAV simulator for specific application such as military and civilian one was presented in [7]. This simulator assumes battlefield environments with theoretical conditions for the design of real systems. The main goal of this simulator is to provide scalability for increased number of UAVs. The simulation results show that common mission can be completed with proposed simulator.

Finally, a new similar UAV simulator was developed for military application. They employ Parrot AR.Drone ver.2.0 platform under Robot Operating System (ROS) framework[8]. The new algorithm was proposed to complete the mission. They include Adaboost Classifier and Pinhole Algorithm to recognize the object with low error rate.

3. Software Component

The software architecture and component for proposed software are illustrated in [9]. In this section, we describe the details for each component and its implementation issues. At First, Figure 1 illustrates the software components and their connections.

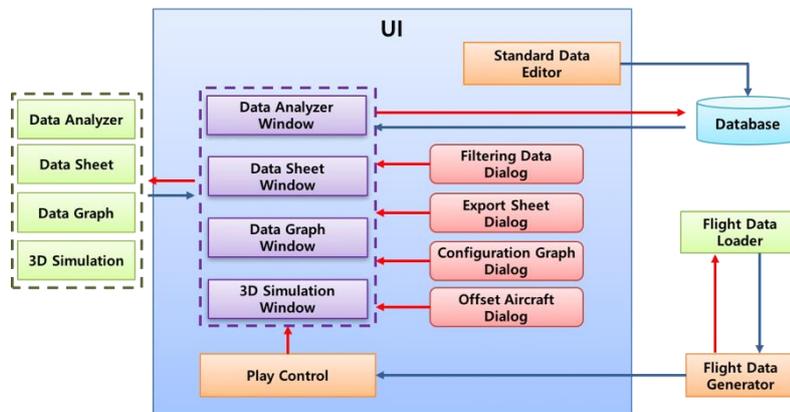


Figure 1. Software Component for Accident Analysis on Drone

User Interface (UI) presents the main display for the software. It has four windows for data analyzer, data sheet, graph, and 3D Simulation. Data analyzer request information for accident to database and get the cause of accident if any matching entry is found. Flight data generator and loader bring flight information from record. In order to update database, Standard Data Editor is given to maintain the accident cases for Drone. Play Control component acquires the flight data and adjusts the configuration to display and analyze data. The data flow in our software is as follows. Flight Data Loader converts the recorded information into simulator format. There are two types information, simulation or analysis information. Play Controller plays a role in create frame for 3D terrain and modeling data. Also, it displays data sheet and graph. This visualization allow scroll to any position while graph makes use of graph for identify the current position. Data analyzer makes use of input data and compare them to recognize abnormal event.

3.1 Flight Data Loader and Generator

Flight Data Loader and Generator accomplish the load of flight information and conduct parsing of them. As a result, simulation data are generated. The data flow is given in Figure 2.

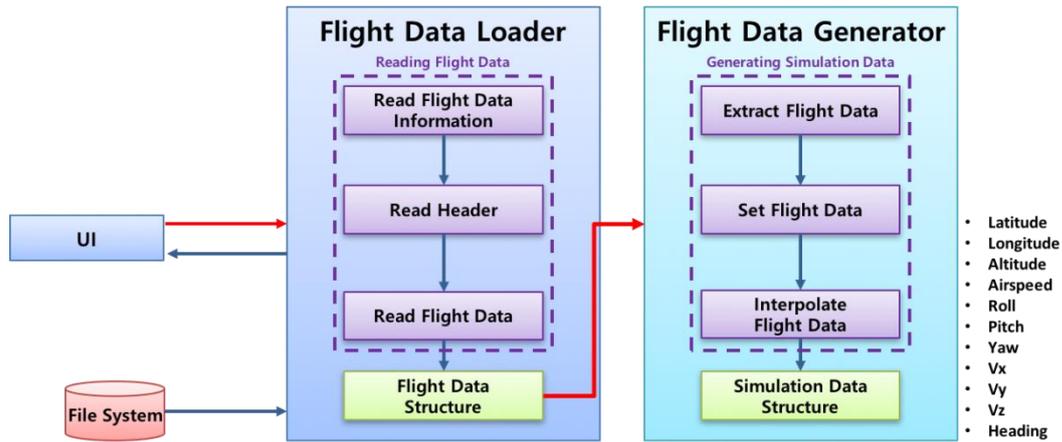


Figure 2. Structure for Load and Generator

In Figure 2, Flight Data Loader reads the recorded flight data and parses corresponding data according to format. Generally, parsing is done in XML form and then we can get the extract many information such as latitude, longitude and so on. This data need to be interpolate according to data processing in the simulator. The flight information consists of Drone model and variables related to it. More detailed, variable name, variable type, output format and size information.

Finally, simulation data structure is generated. With this method, flight information of many Drone model can be converted into simulation data structure if its data format is known.

Flight data generator create flight information for simulator in 50Hz speed through the actual flight information in 8 to 10 Hz. In this phase, the mean value is computed through Spline Interpolation. Also, another concern for generator is the amount of data. If we generate all flight information in 50 Hz speed, total amount of data become lager so it is required to create flight information for simulator with specific variables. They include longitude, altitude, airspeed, roll, pitch, yaw, Vx, Vy Vz, heading.

3.2. 3D Simulation

Next component is 3D simulation which display terrain around the Drone. Also, it illustrates the flying attitude. In addition to this basic display, there are several functions. There are include expanding, shrinking, offset, change of view point. With these functions. the user can identify the flight status and information in easy way. The component of 3D simulation is shown in Figure 2.

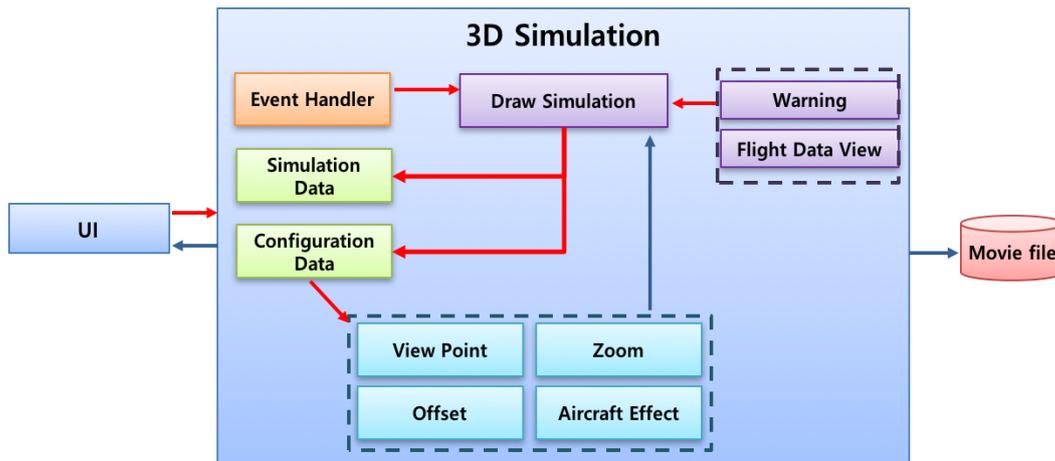


Figure 3. Structure for 3D Simulation

In Figure 3, we can identify the major components for 3D simulation. Below dashed box presents useful functions for display. Also, warning and flight data view are another display way for flight information. This simulation can be recorded on movie file according to user selection.

To display flight status, heading, roll, pitch and yaw information are used. Also, terrain data is loaded according to both longitude and altitude. Other parameters, V_x , V_y and V_z are used to display flight path. Also, their another role is to display the variable on the display or provide some functions to change view point. Recorded movie file contribute to replay the simulation without the proposed software.

3.3. Data Analyzer

The main objective of data analyzer is to provide the possible of accident as well as cause of accident together. The accident is analyzed by comparing cause of accident with actual flight information. When two entries are matched, we can designate the starting and ending time as interval. Also, it can generate the report for analysis result for accident. Figure 4 shows the components for data analyzer.

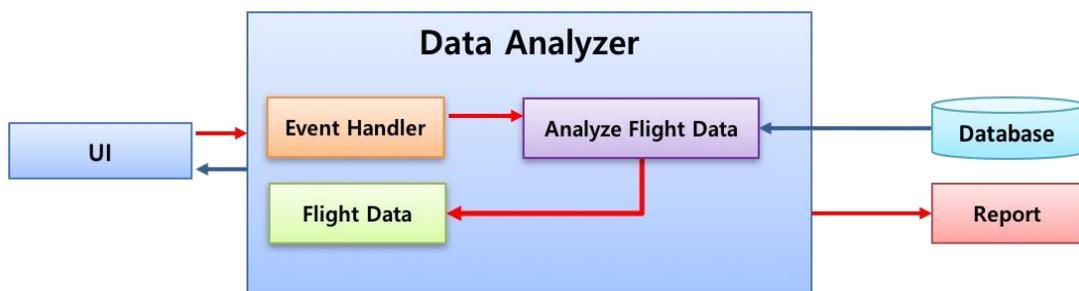


Figure 4. Structure for Data Analyzer

The analyze result is displayed at the play controller so the user can easily identify the position. Also, this result is also displayed with related information while playing the movie file. In addition, we provide separate windows to provide analysis result whenever user request happens.

3.4. Data Sheet

As explained before, data sheet is one of ways for report. In order to create useful data sheet for user, it is demanded to configure parameters such as font or sorting method. Also, how to create cell and fixed column are another parameter for data sheet. If these parameters are set, there are passed to data sheet component. Figure 5 shows the components for data sheet.

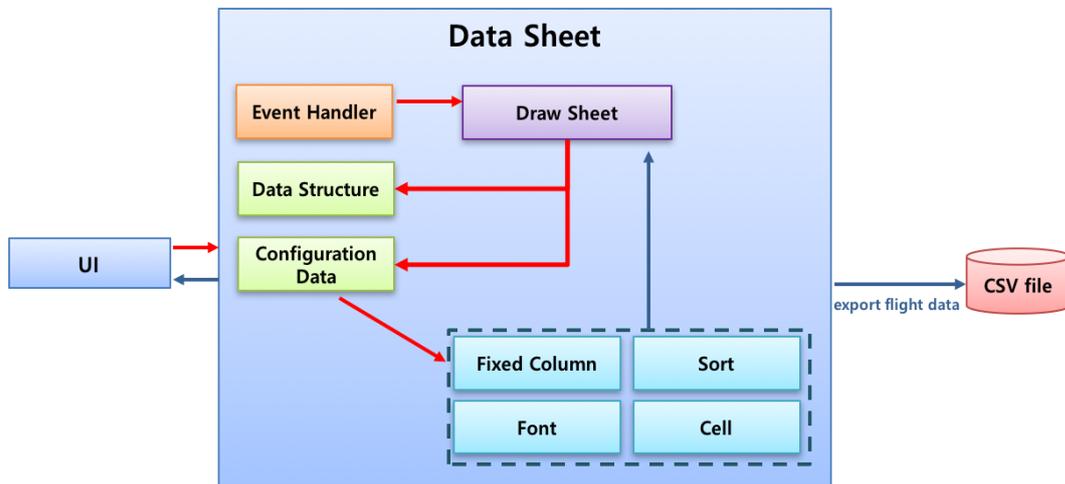


Figure 5. Structure for Data Sheet

After determining the parameters, data sheet can export flight data in CSV files. This CSV files can be converted into several format for display. One of example is graph which is described below.

3.5. Data Graph

Data graph is one of efficient and visual option for data sheet. Similar to data sheet, data graph required configuration step. This is illustrated in Figure 6. For the graph, we need to configure various parameters such as line color, time line and so on. These parameters are easily added or removed. After drawing the graph, this graph can be exported into image file.

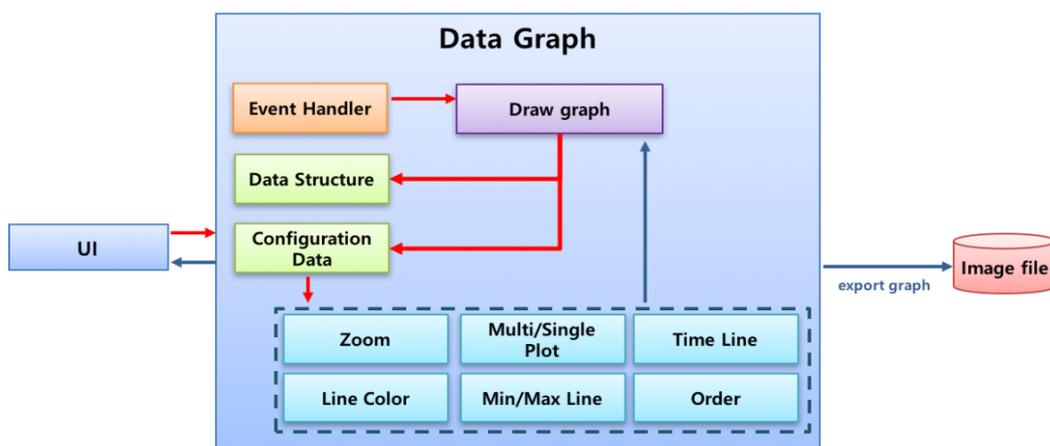


Figure 6. Structure for Data Graph

4. Experimental Results

In this section, we describe implementation issue and challenges. Most of functions are implemented in DLL as module. We use Visual C++ for Integrated Development Environments (IDE). Simulation is implemented through Direct X9 and SQLite3 is for database management. Figure 7 shows the software interface on multiple windows.

The developed software provide multi-document based interface so a user can identify separate windows at the same time. These windows are controlled and integrated by play controller. In case of graph, timeline information is given to user. This is to provide Drone status information at any time. On the other hand, data sheet move scroll at the any point to identify the flight status information. When the specific event happens, play controller marks it and create interval with starting and ending time. If a user selects corresponding interval, it played in a form of box. Also, flight status information is displayed at the same scene to compare them together.

In case of accident analyzer, as we explain before, abnormal data can be easily detected by interactive windows. If this is recognized as accident, starting/ending time, identified variable, possible accident type information is provided in summary. Also, if any user selects one of them, additional detail information is provided to user. By this way, a user can acquire the cause of accident and detail flight status at the accident time. Figure 8 shows two different windows for data analyzer and specific information.

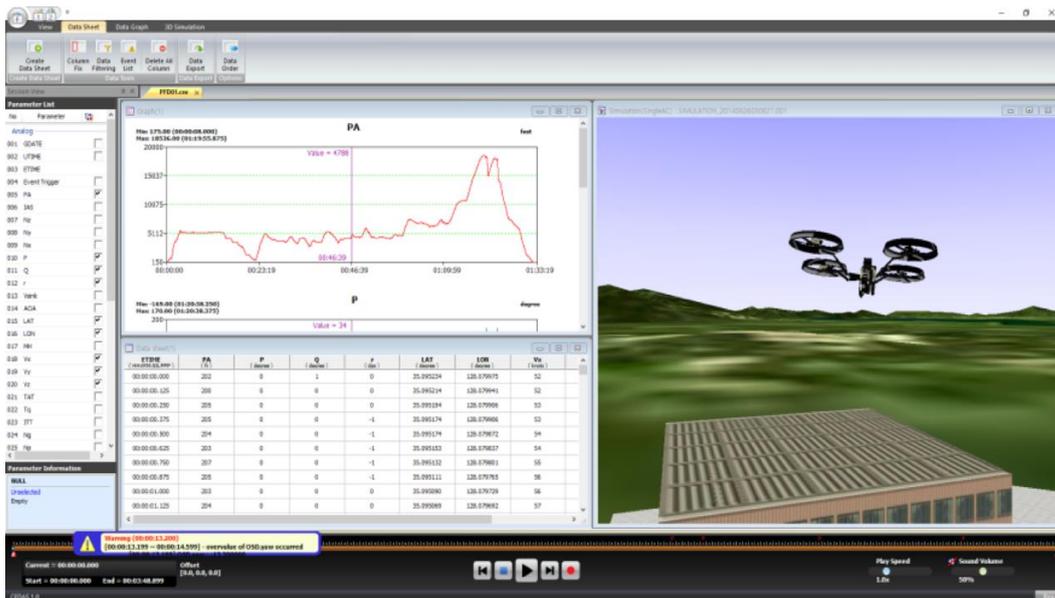


Figure 7. Software Interface

Next is to manage accident cases. This indicates that we need to update and insert new accident cases into database. Figure 9 illustrates the window to insert new case.

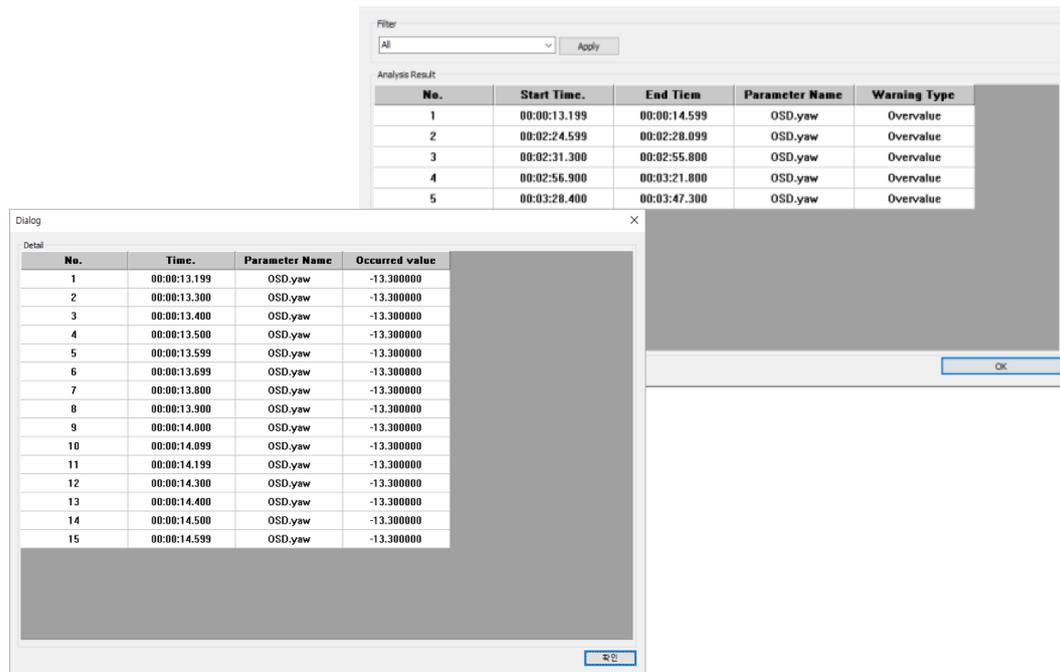


Figure 8. Interface for Data Analyzer

If a user presses [Add] button or selects one existing entry on database, a new interactive window appears. In this window, we can put condition type and type. Also, comparable variable, condition and standard value can be updated by this window.

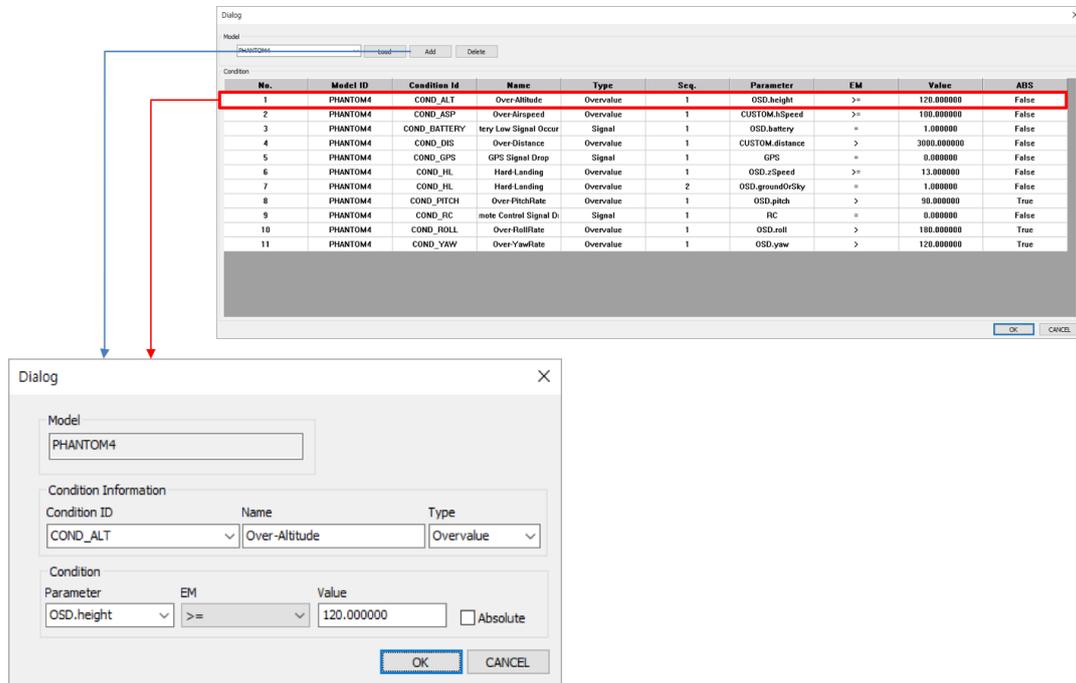


Figure 9. Interface for Accident Case

For the comparison, we concern two cases. One is the case that specific event is not detected through simple comparison. The other is one that there is multiple standard which is based on multiple condition. To achieve this, we apply the AND

logic to employ multiple conditions to find the matching entry. This indicates that we apply OR logic to include two different conditions while AND logic for the same condition. By the help of these implementation, we can find the various cases in efficient way.



Figure 10. Comparison of Analysis Tool

Moreover, Figure 10 shows the user interface for different analyzer which is mentioned in Table 1. As you can see in left side, the existing analyzer provide simple functionality such as displaying flight path on the satellite image or create data sheet or graph in visual way. On the other way, the proposed scheme provides the addressed functions in three dimensional space and useful functions to identify the flight status.

In summary, current simulators provide limited functions and visualization tools for Drone. Thus, if a user need more functions for Drone, they need to develop a new software. But, in this case, two software are not integrated so compatibility can be big problem. This problem is solved by our integrated software. Also, since each component is designed while taking extension, it is easy to include additional functions in our software. Furthermore, another advantage of the proposed software is for data format. Since it is based on general data format,

5. Conclusion

In this paper, we proposed new software to display the flight information through three dimensional simulator and provide accident analysis for Drone. A new software includes not only the most of important functions in existing analyzer tool but also new function. We implement the software and shows the experimental results. Through the experimental results, we demonstrate that the software fulfills the required function and suitability of their operations. Also, since it has extensible architecture for the further function, it can be used to include other implementation.

Related to this work, we will continue the research and implementation to extend database to cloud system. Also, real-time rendering system for terrain will be another research challenges.

Acknowledgements

This work was supported by research fund of Chungnam National University.

References

- [1] Parrot for Developer, <http://developer.parrot.com>.
- [2] 3DR Solo Development Guide, https://dev.3dr.com/?_ga=1.8089138.1469185915.1477376776.
- [3] DJI Developer, <https://developer.dji.com>.
- [4] C. Yun and X. Li, "Design of UAV Flight Simulation Software Based on Simulation Training Method", WSEAS Transaction on Information Science and Applications, vol. 10, no. 2, (2013), pp. 37-46.
- [5] M. David Schmidt, "Simulation and Control of A Quadrotor Unmanned Aerial Vehicle", University of Kentucky Master's Theses, (2011).
- [6] Z. Xing, Y. He and C. Jian, "Design and Implementation of UAV Flight Simulation Based on Matlab/Simulink", International Conference on Advances in Mechanical Engineering and Industrial Informatics, (2015), pp. 190 - 193.
- [7] A. Kasahun Gebremariam, "Graphical Simulation for Unmanned Aerial Vehicle Mission", North Dakota State University Master's Theses, (2014).
- [8] M. A. Ma'sum, M. K. Arrofi, G. Jati, F. Arifin and M. N. Kurniawan, "Simulation of Intelligent Unmanned Aerial Vehicle (UAV) For Military Surveillance", International Conference on Advanced Computer Science and Information Systems, (2013), pp. 161-166.
- [9] M. H. So, S.Y. Kang, K. H. Kim and K.-I. Kim, "Flight Analysis Tools for Drone", Advanced Science and Technology Letters, vol.139, (2016), pp.474-476

Authors



MyungHo So, he is working toward the M.S. degree in Informatics at Gyeongsang National University. He is currently working at Aero Master Corporation. His research interests include avionics software, software engineering and signal processing.



Kyong Hoon Kim, he received his B.S., M.S., and Ph.D. degrees in Computer Science and Engineering from POSTECH, Korea, in 1998, 2000, 2005, respectively. Since 2007, he has been an associate professor at the Department of Informatics, Gyeongsang National University, Jinju, South Korea. From 2005 to 2007, he was a post-doctoral research fellow at CLOUDS lab in the Department of Computer Science and Software Engineering, the University of Melbourne, Australia. His research interests include real-time systems, Cloud computing, and security.



Ki-II Kim, he received the M.S. and Ph.D. degrees in computer science from the Chungnam National University, Daejeon, Korea, in 2002 and 2005, respectively. He is currently with the Department of Computer Science and Engineering at Chungnam National University. His research interests include routing for MANET, QoS in wireless network, multicast, sensor networks and avionics software.