A Study on the Development of Self-driving Platform S/W for Forest Fire Control in Korea

Joon-ho Lee* and Ji-yoon Kim
Korea University, Republic of Korea
juno1238@lh.or.kr, aliasgis@naver.com

Abstract

Two thirds of the land of Korea is covered with forests, and the frequency of forest fires in this country is relatively high. In addition, forest fires are bringing forth serious environment damages and heavy loss of lives. What is more, manpower and equipment mobilized for forest fire fighting often suffer severe damages. In order to minimize such damages, research is being made actively these days for controlling forest fires though IT technologies and unmanned systems such as drones. Considering the characteristics and topographic properties of forests in Korea, forest fire control through unmanned drones may be highly effective. Compared to research on general self-driving platforms, however, few studies have been made on self-driving platforms for forest fire control, which is a somewhat special area. Thus, this study purposed to propose a system for developing the S/W of self-driving robots for forest fire control, to develop and test a simulator, and to suggest an optimized system development model for the future.

Keywords: IOT, spatial information, forest fire, disaster management system

1 Introduction

Recent climate change and artificial deforestation have resulted in frequent large-scale forest fires throughout the world. In Korea, 384 forest fires take place and they burn 631ha each year on the average (during the period from 2005 to 2014), and the frequency is 26% higher than the 10-year average (389)[8]. Forest fires entail environmental damages and loss of lives, and the restoration of forests require social costs. Thus, various studies are being made throughout the world to prevent the damages and spread of forest fires. With the recent advance of IOT technology, there are an increasing number of attempts to utilize drones and sensors in forest fire control. Among different areas of IOT technology, self-driving applied usually to unmanned vehicles and drones aims at safe and efficient unmanned driving by combining S/W with H/W such as cars and aircrafts[5]. It is highly useful particularly for monitoring areas or topographies hardly accessible to humans or for preventing disasters, and therefore, IOT technology for these purposes is researched actively in many countries throughout the world. In a general-purpose self-driving platform, the most crucial elements are geographic information technology and sensor information acquisition technology, and these functions must be included in the S/W platform. Particularly for carrying out the mission of forest fire control in peculiar forest spaces, the platform should ensure S/W functions related to information acquisition and analysis and many special situations expected. For the utilization of self-driving technology in forest fire control, moreover, we need to develop S/W technology that can analyze the topography, stand composition, climatic characteristics, etc. of forest environment holistically and intuitively. In Korea, moreover, relatively detailed spatial and analytical information on forest

---

IT CoNvergence PRActice (INPRA), volume: 4, number: 3 (September 2016), pp. 9-19

*Corresponding author: Gae-sung Branch, Korea Land & Housing Corporation 3 seongnam-daero 54 beon-gil, Bundang-gu, Seongnam-si, Korea 13637, Tel: +82-(0)10-9020-1375 fires and forests is open to the public, so it is possible to develop and test platforms and various simulation situations. This study aims to suggest a strategy to build an efficient S/W platform by proposing a design and configuration of self-driving S/W platform functions for forest fire control using forest-related open data. In addition, this study implements some of the proposed functions to assess the implementability of the S/W platform, and discusses its limitations and solutions.

2 System and Platform

2.1 The functions of self-driving S/W platform for forest fire control

In case of a self-driving vehicle, the system and platform are designed for self-driving in urban areas or roads. Forests and mountains are different from urban areas in topographic factors and network environment. When special situations such as forest fires are considered, what is more, there have to be functional differences. Nevertheless, the basic functions and concept of self-driving S/W platform are not different. Table 1 shows the basic functions and specifications of self-driving platform.

Table 1 shows areas where research is active for self-driving vehicles, and is a good example for the module configuration of a self-driving platform. They are essential functions and basic required specifications in using a self-driving platform for forest fire control. As mentioned above, however, a self-driving platform for forest fire control needs to consider the topographic and ecological properties of forests, obstacles not found in ordinary driving roads, and driving environment in the special situation of fire, and with these factors, it has many differences from other types of self-driving S/W platforms. We can find a good example in the state-of-the-art firefighting robot equipped with abilities to detect and extinguish forest fires by itself developed at the University of Magdeburg Stendal in Germany. This robot, named ‘OLE’ [Off-road LoEscheinheit (off-road firefighting equipment)], is a firefighting robot specialized for forest fire control. This is 1.2m long and 0.6m wide, having a body similar in size to rescue dog St. Bernard, and has a water tank and a powder fire extinguisher inside for extinguishing fires. The biggest characteristic of OLE is that it is an intelligent self-driving robot. When a forest fire occurs, it is not operated to extinguish the fire by humans in a remote place, but it determines the occurrence of a fire, accesses the site, and controls the fire. For this reason, OLE does not have any camera, which is essential equipment for most of robots. Instead, it surveys things around using its GPS transmitter/receiver, infrared sensor, and heat detection sensor, and finds the site of forest fire. It is also equipped with a high-performance bio sensor to detect a fire by smell in case the origin of fire is blocked by obstacles. With these sensors, OLE can detect a fire up to 800m away. With a few OLEs deployed in the forest, the firefighting authority can prevent forest fires without regular patrolling or installation of hundreds or thousands of surveillance cameras. What is more, GPS plays the function of keeping OLE, which is estimated to cost 125 200 thousand dollars, from being stolen.[4]

Based on this background information, a self-driving S/W platform for forest fire control is required to have characteristics as follows. It should be able to integrate sensor data and images and to move quickly to the spot of first forest fire. For these functions, it should be able to analyze integrated sensor information and CCTV data and to survey and predict possible spots of forest fire accurately. Second, it should be able to use topographic information for safe and fast travelling in a forest, and have an algorithm for avoiding collisions with obstacles such as miscellaneous trees in the forest. Forest fire control requires basic topographic information on roads, water systems, etc. in the forest. As the Korean government has built extensive basic forest geographic information including forest type maps and opens it to the public according to its policy, it is easy to obtain relevant information. In order to move quickly to the site of forest fire in a forest, path planning should be used, and the obstacle avoidance function is also essential to run through various types of obstacles including stickers and rocks in the forest. Particularly because Korean forests have an increasing number of trees through over 40years’ afforestation, it is

Function	Defition
Radar-based driving situation recognition module	Radar-based and LIDAR sensor-based recognition and detection technology to measure the distance and volume of various target objects (or obstacles on the path of driving) in driving environment and recognize the accurate distance and spatial information of the objects.
Image sensor-based information-integrating driving situation recognition technology	Image sensor-based recognition and detection technology to process information on the form and distance of traffic lanes, signs, vehicles, motorcycles, etc. for maintaining the driving path, changing traffic lanes, supporting merging and branching, guiding parking, automatic parking, etc.
V2X(V2V & V2I) communication technology for scalability, versatility, and security	Reliable V2X communication module design technology to recognize situations surrounding a vehicle by integrating infrastructure and vehicle sensor information using V2X communication technologies such as V2V (Vehicle to Vehicle) and V2I (Vehicle to Infrastructure)
Digital map technology including information on roads and topographic properties for self-driving	Digital map generation technology for ADAS (Advanced Driver Assistance Systems) having information on roads and topographic properties for self-driving that enables the maintenance of traffic lane and the prediction of driving road using data on the traffic lanes, roads, and grade of a over 1km long segment ahead of the driving path
General-glass high-precision hybrid positioning technology	Technology to estimate the position, driving direction, and speed of a vehicle using low-end DGPS in self-driving environment by integrating DGPS, digital map, IVN (Inter-Vehicle Network) and surround sensor information
Smart actuator technology in consideration of fail-safety	High-reliability smart actuator design technology reflecting redundancy and fail safety for the safe control of a self-driving vehicle through system risk analysis and failure analysis
Self-driving HVI (Human Vehicle Interface) technology in consideration of driver receptivity	Next-generation HVI technology improving driving safety, convenience, and receptivity (relief of anxiety) through the derivation of UX scenarios and the development of optimal UI by driver's gender/age through analyzing/evaluating information on the driver's (including the transportation vulnerable) characteristics, tendency, and driving condition, and the internal/external situations of the vehicle in actual self-driving road environment
Driver monitoring technology for safe self-driving	Driver monitoring technology for making self-driving control strategies by processing information on the driver's condition (load, fatigue, etc.), emotion, tendency, and intention and on situations surrounding the vehicle using various interfaces including video, audio, and haptic ones in vehicle environment
Next-generation IVN (In Vehicle Network) platform and integration controller technology	Next-generation IVN platform and IVN-based integration controller design technology that includes the concept of redundancy and adopts E/E architecture of new concept for guaranteeing smooth self-driving of vehicle and the driver's safety
EDR(Event Data Recorder) technology for finding the causes of accidents in self-driving	Technology for saving, protecting and transmitting image information inside and outside of a self-driving vehicle, surround sensor information, and IVN (Inter-Vehicle Network) in real-time for the safe operation of self-driving vehicles and finding the causes of accidents

Table 1: Self-driving functions [6]

becoming more common for a surface fire, which burns only the litter layer on the ground, to develop into a 'crown fire,' which burns whole trees and causes a more severe damage, and as a result, a forest fire itself produces many obstacles. Therefore, such an obstacle avoidance algorithm needs to be mounted on the S/W platform. Path planning algorithm, which is a methodology for this, generates the optimal path from given maps and other types of information to reach a desired destination at the minimum cost while avoiding collision with fixed obstacles.[10]

Third, the S/W platform should be able to optimize forest fire control and driving using databases of past forest fires and forest ecosystems. The latest artificial learning techniques such as deep learning are believed to be applicable in the area of forest fire control. Fourth, as it is difficult to utilize networks in a forest due to the peculiarities of forest spaces, we need to synchronize system data and network communication. Synchronization is causing events to occur or operate with coincidence for the operation of a system. It is used often in computer systems (communication between functional modules, data transmission between memory and CPU), networks, GPS, etc. Once the purpose of synchronization is defined, the system designer decides the method of synchronization. Synchronization can be achieved through H/W or S/W. It is necessary for smooth transmission and reception of data in the field of communication. Because required elements are different depending on data transmission method and rate, the designer designs the procedure and makes rules. Synchronization is used not only for long-distance communication but also short-distance one within a computer or communication system. Synchronization is essential for the functions of a self-driving platform operating inside a forest and for its interoperation with systems outside the forest. It is because the unusual situation of forest fire may hinder the operation of a self-driving platform, and the monitoring and control of its operation, and the integration of information from the outside of the forest such as weather information is crucial for interoperation.

2.2 Design of communication and interoperation of self-driving S/W platform for forest fire control

A S/W platform for forest fire control is distinguished from other types of self-driving platforms due to difference in the environment of interoperation. A S/W platform for forest fire control needs to support safe driving, forest fire restraining and extinguishing, forest fire situation monitoring, and the transmission of information on forest fire situation to the outside. For the optimization of the moving path of the self-driving platform, what is more, the self-driving platform and the monitoring platform should be separated from each other as client and server layers. Even if the server and client are disconnected from each other due to an external situation, the self-driving platform, as a client, should operate independently. Server-client communication is designed based on HTTP, and communication between internal equipment such as sensors and the S/W platform is implemented with RS/232, RS/485, etc. These communication protocols are serial communication interfaces used commonly when system engineers control or monitor machines. As sensors, terminals, etc. inside a platform have to communicate with and control one another, easily controllable and proved protocols are used. Sensor information inside the platform is transmitted to the server controlling the driving platform through a transmission module.

The transmission module generates prior information necessary for safe driving such as path information, forest ecosystem information, weather information, and spatial information, and transmits it to the driving module. With the received information, the driving module obtains data about the surrounding circumstances of a forest fire area from sensor and camera information, and control the forest fire through safe and continuous self-driving. Table 2 below defines the functions and roles of necessary sensors under a self-driving platform.

In a forest fire situation, data generated by these sensors is integrated and shared with the server through communication for the monitoring of the situation and for the safe driving of the self-driving platform. What is more, sensor information enables optimal planning for avoiding obstacles caused by a

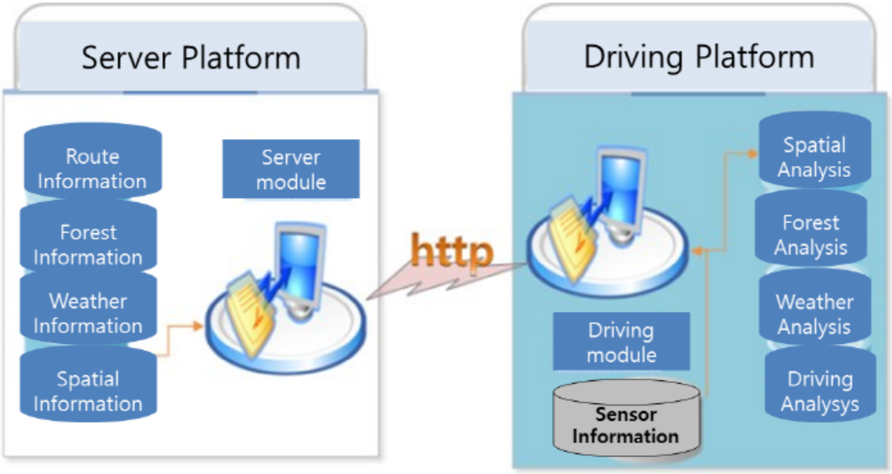

Figure 1: Information transmission of driving platform

Sensor for self-driving	Role
Temperature sensor	Sense the temperature of the ground surface, and estimate the damage pattern and accessibility of a forest fire
Wind direction sensor	Sense the wind direction in a forest fire area and predict the pattern of fire spread
Barometric pressure sensor	Interpret weather and weather data by reading the flow of air
Speed sensor	Maintain safe driving by reading the current speed of the self-driving platform.
Position sensor	Indicate the current position of the self-driving
Tilt sensor	Guide safe driving by checking the gradient of the self-driving platform

Table 2: The roles of built-in sensors for forest fire control

forest fire, and it is possible to find the optimal path for forest fire control by comparing forest fire information generated by different servers. What is more, a safe driving path can be set by integrating existing forest spatial information with forest fire information from the sensors. Because a forest has many bends and curves as well as uneven ground surfaces, a self-driving platform operates a gyro sensor for driving in a right posture, and keeps checking the gradient of the gyro sensor under S/W platform. By monitoring gradient and analyzing the measurements of the gyro sensor, the platform can issue commands for maintaining balance. In addition, it can monitor the normality of driving through communication and interoperation.

2.3 Mounting a self-driving module of forest fire spread prediction algorithm

Forest fire researchers have long studied the path and spread speed of forest fires for forest fire control. With the recent advance of geographic information technologies, various trials and studies are being made including the 3D visualization of the results of forest fire spread prediction. In Korea, S/W was developed in 2009 for predicting the spread path and damage of forest fires in real-time. As a 3D forest fire spread program predicting the path and intensity of forest fires by time, this S/W can process up to 2.5 times faster than existing prediction programs[3].

Forest fires develop in complicated patterns involving various factors including topography, tree type, and weather conditions such as wind speed, wind direction, and humidity, and therefore, it is very important to analyze spatial information in real-time by hour. Spread prediction is crucial because forest fire spread path, fire control line, damage segment, etc. can be derived from the prediction. Forest fire spread prediction algorithm uses cell-based gradients and landforms and calculates weights among cells. As these cell-based operations consume a lot of computer resources, real-time calculation inside a self-driving platform appears infeasible. Therefore, a self-driving platform for forest fire control needs to have spread prediction scenarios for different weather and environmental conditions built from existing information. The server plays the roles of generating spread prediction scenario information and sending the information on a client platform's request. The flowchart below shows the algorithm for selecting a forest fire control scenario under a self-driving platform.

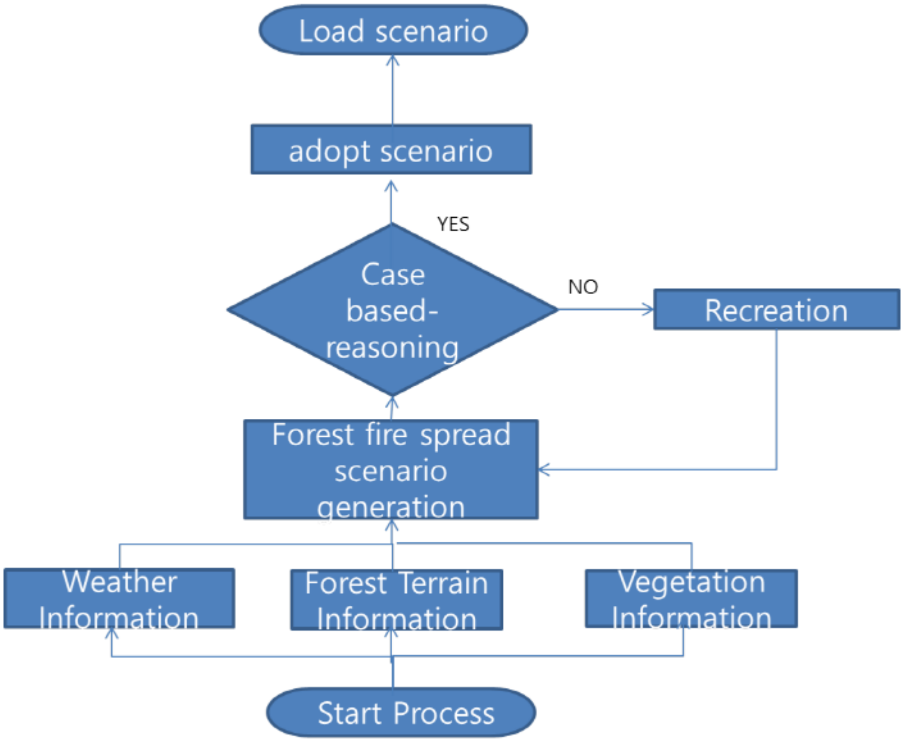

Figure 2: Method of mounting forest fire spread scenarios

In selecting a scenario, necessary information is provided for basic analysis of forest fire spread and spread prediction scenarios are generated, and then, forest fire spread information from similar environments in the past is compared space-statistically. For the statistical comparison, forest fire spread information to be compared is generated in the form of space cell grid, the cells are compared, and similar information is utilized. Raster data are used in a GIS system for displaying a continuous geographic phenomenon or information not divisible into vector data[1].

There are several methods for measuring similarity between spatial cells, but GIS-based analyses usually utilize the Euclidean method. Because this method assesses similarity between two cells using the distance between the two, however, it has a difficulty in analysis for predicting forest fire spread, in which direction is important, and therefore, we need to find a method of using cosine similarity that emphasizes direction. Cosine similarity is also used to measure cohesion among clusters in the field of data mining[9].

Forest fire spread prediction scenarios are generated under the server platform, and are mounted on an actual driving platform. The data mounted on the driving platform need to be slimmed down for a high computing speed. Because original data from GIS analysis are relatively large in volume, they should be reformatted for fast analysis in driving environment. For data reformatting, we adopted Geojson, which is a lightweight format. Geojson is one of open data formats for geographic information, and it uses the grammar of Java Script[7].

```
{ ”type”: ”FeatureCollection”,
{ ”type”: ”Feature”,
”geometry”:{ ”type”: ”Polygon”,
”coordinates”:[[ [100.0, 0.0], [101.0, 0.0], [101.0, 1.0], [100.0, 1.0], [100.0,
0.0] ] ]},
”properties”: { ”prop0”: ”value0”, ”prop1”:{”this”:hat”} } } ] }
```

Table 3: An example of Geojson

In the conversion process, raster grid cells are converted to the polygon type of Geojson. For the conversion, JSON data were processed using GDAL/OGR, which is an open source GIS library. Table 4 below is an example of scenario data to be mounted under a driving platform.

Max X	Max Y	Min X	MinY	Cel Value
125.46556	37.74655	125.464511	37.740042	1
125.464456	37.74655	125.434511	37.7340042	1
125.466500	37.74655	125.464511	37.740042	0.8
125.436454	37.743111	125.46777	37.745047	0.8
125.436450	37.74211	125.46777	37.745047	0.8
125.446454	37.74111	125.47777	37.746047	0.8
125.436454	37.743111	125.46777	37.745047	0.8

Table 4: A sample of forest fire spread prediction scenario data

The format of these data is that used in Forest-R2, a project to develop an open source-based self-driving SW platform for forest fire control. This sample scenario data use the cosine similarity algorithm, and target the area of Mt. Surak in Seoul. The data designed as above are for predicting forest fire spread paths from the current position of the driving platform. When the driving position comes within the range of latitude and longitude according to existing data, the platform reads the forest fire spread prediction values and refers to them in selecting a path for driving and forest fire control.

2.4 Configuration of self-driving S/W platform module for forest fire control

When the discussions are summed up, a self-driving S/W platform for forest fire control has characteristics as follows. First, because sufficient computer resources cannot be mobilized on the network due to the special situation of forest fire and the topographic peculiarities, we need configuration that synchronizes data and path information for forest fire control between the network and the driving platform. Second, different from self-driving platforms for general vehicles, a self-driving platform for forest fire control should be able to recognize and analyze various types of information necessary for forest management and information on environmental situations. Third, a self-driving platform for forest fire control should provide functions essential in general self-driving platforms such as collision avoidance and optimal path selection. Figure 3 shows the configuration and functions of a self-driving platform module for forest fire control designed in consideration of these conditions.

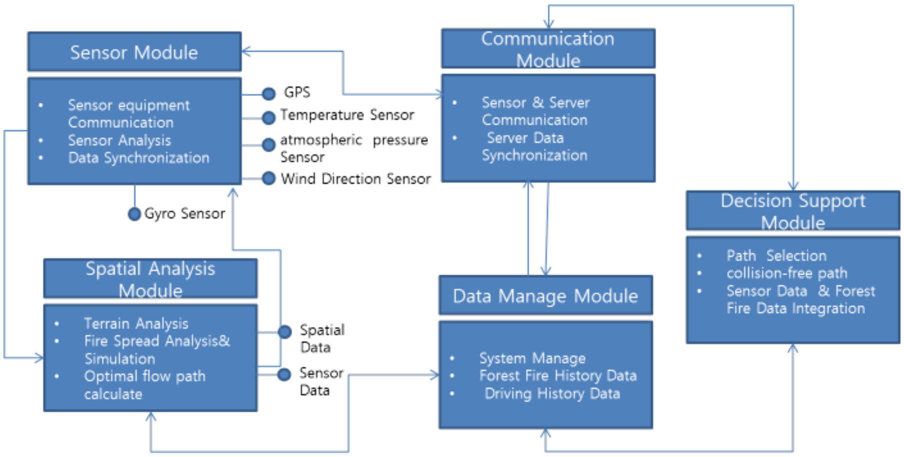

Figure 3: Configuration of a self-driving platform module for forest fire control

The module configuration above has system architecture that modularizes and integrates individual functions and the H/W platform. It componentizes individual functions into sensor, communication, spatial analysis, and data management. The data module with a database is used for inter-module interface and system connection. The database stores and manages all the data of the driving platform, history of system interoperation, and forest fire information. The data module plays the role of interface for retrieving and storing necessary information from and to the database, and working together with the communication module, it transmits and generates data for supplying information to functions and making decisions. In the information integration and decision-making system, information is analyzed for actual self-driving functions and H/W control is implemented. For this, techniques such as machine learning are used for safe and fast control. Based on the module design above, prototype simulator program ForestR2D2 was developed and used to test the adequacy of the module design and its applicability in actual driving and controlling works.

At present, orestR2D2 is being developed in .NET development environment under the Windows platform. This is an open source project started in March, 2016. The client was developed with C# WinForms, and the server was developed in asp.net environment.

This project will be open first through GitHub in June, 2016. Through this project, we could test the functions and module architecture of the self-driving S/W platform for forest fire control. The function under implementation at present is for monitoring GPS signal position and setting a path using predesigned scenario data, and this enabled function tests. GPS signal used in this project was generated as a log file in the format of NMEA 0183, and was interoperated with the communication module. NMEA 0183, which is commonly called NMEA, is a standard for transmitting information of time, position, azimuth, etc[2].

Test data were generated as defined in this standard, and a test was conducted by sending the test data to a UDP communication server and ForestR2D2 of broadcasting type developed in this study, and checking whether it is possible to monitor the moving of the self-driving platform to the preset destination in the map of the computer screen. The reason for testing through UDP communication in ForestR2D2 is that we assumed a self-driving platform in which information exchange and sharing is possible through communication between multiple driving simulators and the server. The experiment implemented communication among five driving probes for 24 hours, and tested the normality of driving in a forest fire area and the normality of data transmission and reception by hour. In addition, driving

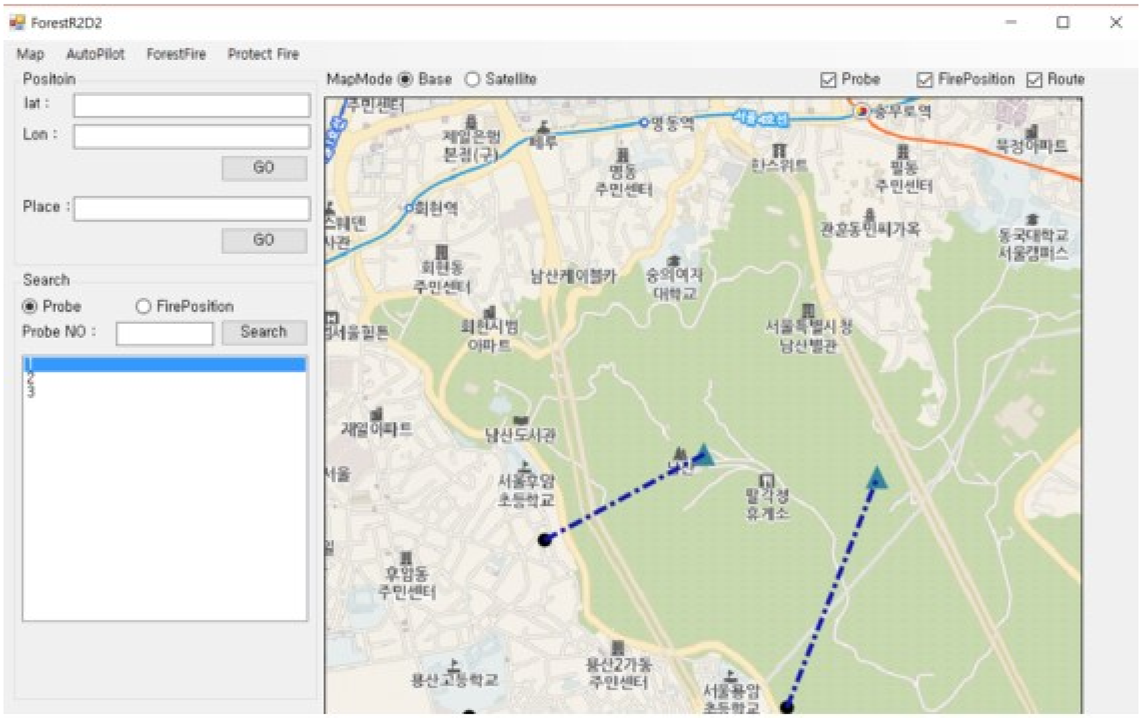

Figure 4: Self-driving platform simulator program ForestR2D2 for forest fire control

terminals were added in the middle of the experiment, and their normal operation was tested. In the test, the specifications of the client were CPU i7-2.4ghz and memory 8GB. On the other hand, the server was configured with Intel(R) Xeon(R) CPU E5606 @ 2.13GHz, and memory 32GB.

Number of probes	Server error	Client error (symptom)
5	0	1
10	0	1
15	Low processing speed	Driving stopped
20	Low processing speed	Driving stopped

Table 5: Results of simulator testing

According to the results of the test, the more the probes are, the more communication errors and abnormal operations took place. This indicates the need of cloud environment and distributed processing in actual driving where a lot of computer resources are mobilized.

3 Conclusions and future research

In gear with the advance of highly sophisticated automatic vehicle systems and the development of artificial intelligence technologies such as deep learning, self-driving platform is one of rapidly developing areas in terms of market size and technology. Forest fire or disaster control using a self-driving platform will be greatly helpful for reducing the loss of lives and properties. In addition, because forest fire control entails many casualties and damages in Korea, the use of self-driving platform in forest management can provide a solution for these problems, and the speed of its development may be accelerated by IT infrastructure and the availability of public data. However, self-driving platforms for forest fire control are functionally different from ordinary vehicle driving platforms because they have to reflect the topographic properties of forests and the special environment of fire. The functional differences again require different module configuration and design of the S/W and H/W platforms. Through this study, we suggested module configuration and system architecture optimized for self-driving S/W platforms in forest fire control areas. Given the spatial peculiarity of forests and the special situation of forest fire, the platform should synchronize data and systems on the network for self-driving, and maintain smooth communication of spatial information with sensors. As to spatial information, a self-driving S/W platform for forest fire control utilizes GIS-based topical forest maps and forest fire spread scenario data as key information for self-driving. In order to test if such a system configuration is plausible, this study developed a test S/W program named orestR2D2, and tested some of its functions. In order to upgrade this study and its outcomes, we need to implement all of the S/W functions and integrate them into a system. S/W integration requires not only S/W functions but also self-driving H/W and middleware for interoperation with the S/W. What is more, the system should be tested in environment similar to an actual forest fire situation. For such a driving test, we need to conduct S/W engineering tests and to standardize test methods and test conditions.

References

[1] Raster data. QGIS oneline 2.2 manual. `http://docs.qgis.org/2.2/ko/docs/gentle_gis_introduction/raster_data.html` [Online; Accessed on September 1, 2016].

[2] N. E. M. Association. Nmea 0183 standard. `https://www.nmea.org/content/nmea_standards/nmea_0183_v_410.asp` [Online; Accessed on September 1, 2016].

[3] Donga science. The wildfire spread is predicted in real time, March 2009. `http://news.dongascience.com/PHP/NewsView.php?kisaid=20090325200000015814&classcode=011717` [Online; Accessed on September 1, 2016].

[4] Y. ha Jeong, H. woon Park, S. jin Lee, and M. cheol Won. Development of a Navigation Control Algorithm for Mobile Robots Using D* Search and Fuzzy Algorithm. *Transaction of the Korean Society of Mechanical Engineers A*, 34(8):971–980, 2010.

[5] Hankook-Ilbo, March 2015. `http://www.hankookilbo.com/v/51d4aef0d34b4595a4732f060560c5b5)` [Online; Accessed on September 1, 2016].

[6] J. K. Lee. Outlook and Trends in Autonomous Cars. *Fusion Weekly TIP*, 4:1–8, August 2015.

[7] L. Richardson, M. Amundsen, and S. Ruby. *RESTful Web APIs*. O'Reilly Media, September 2013.

[8] K. F. Service. Forest statistics, 2015. `www.index.go.kr/potal/stts/idxMain/selectPoSttsIdxMainPrint.do?idx_cd=1309&board_cd=INDX_001` [Online; Accessed on September 1, 2016].

[9] A. Singhal. ”modern information retrieval: A brief overview”. *IEEE Data Engineering Bulletin*, 24(4):35–43, 2001.

[10] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison-Wesley, ISBN 0321321367, chapter 8; page 500, 2005.

Author Biography

Joon-ho Lee is a manager of Korea Land and Housing Corporation. Previously, he developed the Korea Land Information Systems (KLIS), Land Use Regulation Information System (LURIS) of the Ministry of Land, Infrastructure and Transport. Also, he worked in GIS Research Center, Korea Research Institute for Human Settlements (2000-2002). He is now a Ph.D. candidate in Environment Eco Science at Korea University. His research interests include National Spatial Data Infrastructure (NSDI), Urban planning information system, and GIS.

Ji-yoon Kim is a Software Developer at AKGEO.Inc in Korea. In 2015, he obtained his Ph.D. from Environment Eco Science at Korea University. He worked as Geography Information System Engineer and Spatial Analyst for 15 years. He developed S-100 standard based on e-navigation system while recently developing the cloud based service of IoT platform. His major interests are in e-Navigation, GIS, remote sensing, climate change, and spatial analysis.