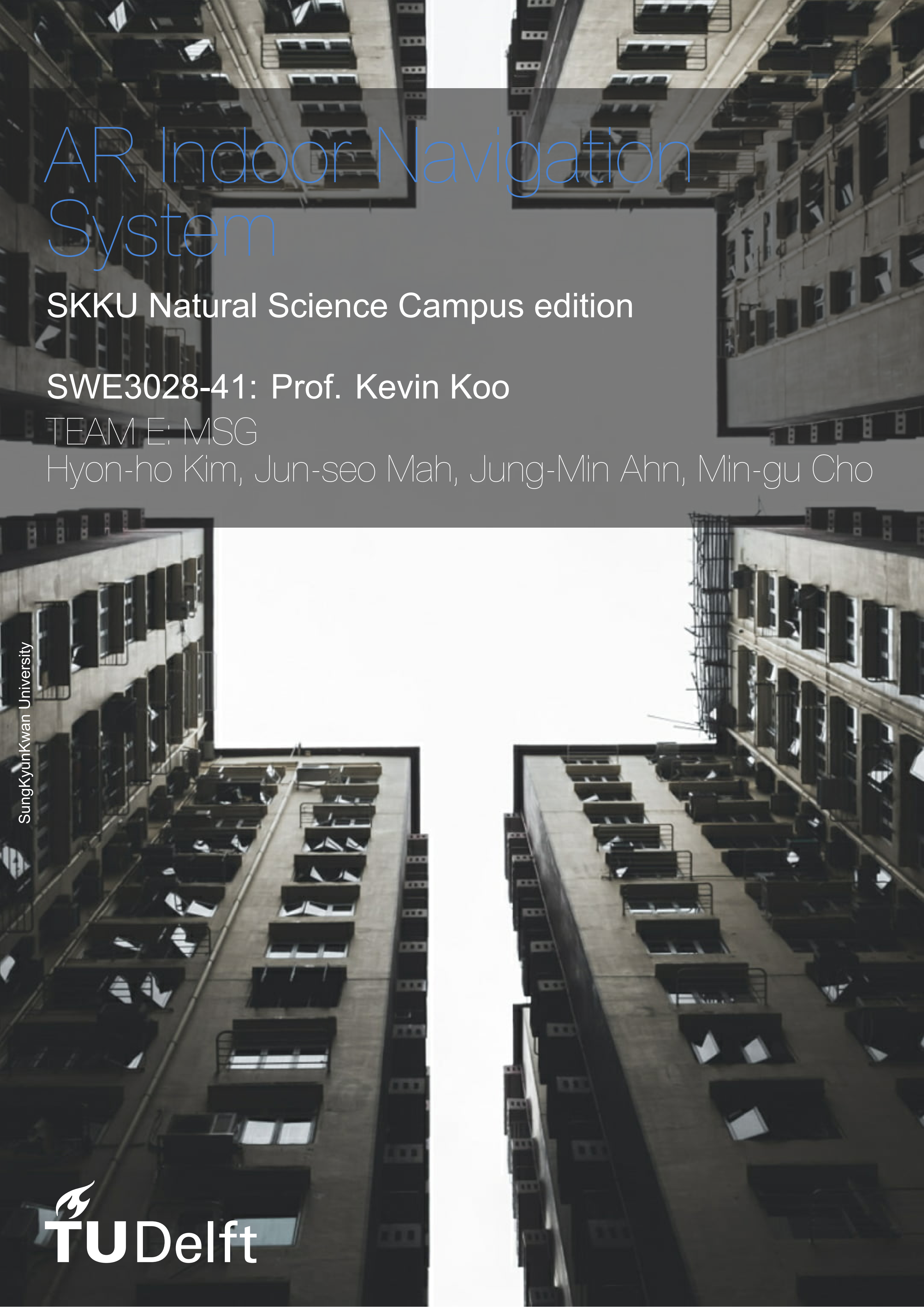

AR Indoor Navigation System

SKKU Natural Science Campus edition

SWE3028-41: Prof. Kevin Koo

TEAM E: MSG

Hyon-ho Kim, Jun-seo Mah, Jung-Min Ahn, Min-gu Cho

SungKyunKwan University

AR Indoor Navigation for SKKU Natural Science Campus

TEAM E: MSG
Hyon-ho Kim, Jun-seo Mah, Jung-Min Ahn, Min-gu Cho

March 25, 2022

1 Abstract

The content of this document shows the proposal of TEAM E: MSG's answer in indoor navigation method. The team will be implementing UNITY, AR MARKER, and AR ANCHORING, TRIANGULATION CALCULATION within the app to help user navigate through "Benzene-ring" building in Sungkyunkwan university within 12 week period.

2 Introduction

With the advancement in technology, not only are the buildings becoming bigger, but they are becoming more complex. This had lead the users of the building confused and losing their bearing in the building. Such results are either one or both fruition of bad architectural design category; over-complexity and over-similarity.

The most infamous structure suffering from over-complexity in Korea is "COEX" department store. The latter category of bad architectural design is the infamous "Benzene-ring" building, situated in SungKyunKawn Natural Science Campus. Our team have decided to tackle the latter categorical problem, more specifically SungKyunKwan building, to help user navigate in confusing indoor more easily.

Figure 1: Lost University Students

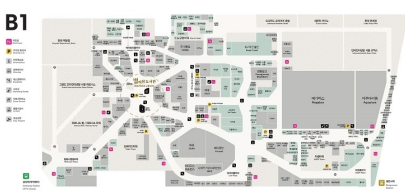

Figure 2: COEX Building

Figure 3: Benzene-ring Building

3 Objective and Proposed Solution

To tackle this problem, the team set the goal of ”building an ubiquitous Indoor Artificial Reality(AR) Navigation System for the Campus”.

3.1 General System Flow

First, consider a situation where the user turns on navigation system. The software must recognizes the initial location. It is expected that most of the time when the system is initialized, the user is trying to either enter the building through the main entrance or exit the class room. By using the door-plate placed next to each room as a marker, it is not difficult for the system to recognizing the starting point in most of the cases.

When the starting point is recognized and the user starts moving, the movement is detected and reflected through smartphone’s gravity and acceleration sensor. This, however, cannot perfectly detect the user’s movement, resulting in constant error. When this error accumulates, of course, a problem occurs in navigation guidance. Due to the nature of AR, which outputs content based on the relative position of the user, an error in navigation guidance would result in AR outputting wrong location content.

To correct this error and synchronize the navigation position with the user’s current position. The team will be designating and taking scanned images of feature points within the building, and set it up as AR anchor. When the user’s phone camera approaches this feature point, the system will adjust the user’s current position through triangulation calculation by comparing relative position of phone’s place when it recognized the feature point and saved anchor’s position in system database.

3.2 Technical Background

Though Many functions would be integrated within the system, this subsection will explain the main integral part of the Indoor Navigational System.

3.2.1 AR Environment Building

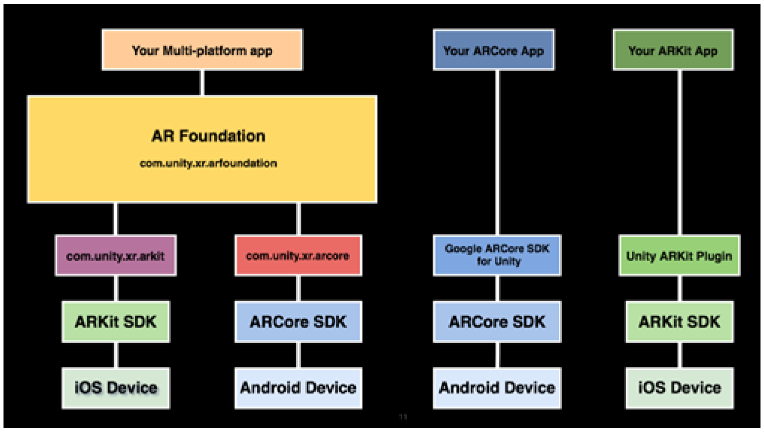

Figure 4: Different AR SDK

Since the entire system will be sitting based on the AR Environment framework, the choice of using what kind of framework is important.

As for the AR environment implementation System Development Kit(SDK), AR Core provided by Google or AR Kit supported by Apple is generally used. AR Core supports both Android and IOS, and AR Kit supports IOS.

In this project, the team will be using the AR foundation framework supported by Unity; this AR foundation includes unique Unity functions in addition to core functions such as Apple's AR Kit, Google's AR Core, and Microsoft's HoloLens through an integrated workflow.

It is useful to build and manage AR environments such as output of AR images that are natural to the real environment through detection of and depth, device tracking, etc. It also provides information on real space when necessary for our navigation to work. With additional advantages such as being able to be implemented on various platforms such as Android and IOS with one working version.

The Unity environment work tool also allow us to visually check and edit 3D model based mapping, it gives intuitiveness and bring convenience to the work. In addition, thanks to its great compatibility, not only can it use libraries such as openCV for image preprocessing in AR marker, but it can be used as an engine for front-end development.

3.2.2 Mapping

Mapping is the process of creating a virtual map of an indoor space and inputting information data about key places on the map.

There are two main methods we explored for mapping, the first is feature based mapping. This is a method that has been mainly studied in the field of autonomous driving robots. However as written in in Hurdle section(add section connection), this system in its entirety is not viable to our team.

However, mapping a place with a landmark that is still clearly distinguished from the surroundings is very useful for localizing the user's location for our project. According to a paper published in 2020, if the AR anchor is installed based on this recognized cloud point, later when the system undergoes trilateration process based on the anchor, a fairly accurate distance from the anchor can be known.

For the entire mapping of the building, we plan to use the actual building schematics, containing height and width of the room and corridors. Though such schematics method could be easily recreated by measuring each length of the room and corridor due to its inexpensiveness, manually measuring distance by ruler add error range within the system, leading to navigation error. Fortunately, the university management team, has promised to provide the team with the entire building blueprint. Thus, eliminating the error range within the system caused my incorrect measurement.

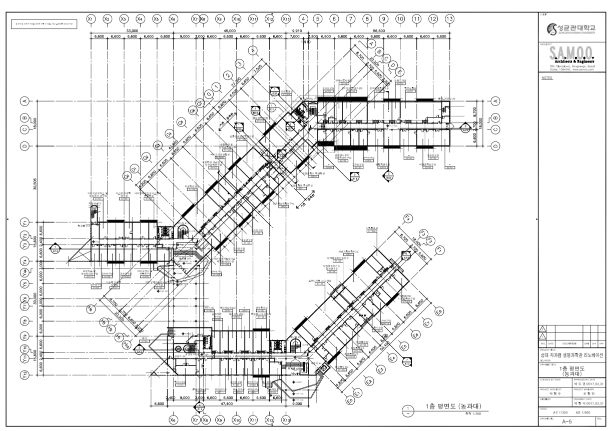

Figure 5: Benzene-ring Building schematics

3.2.3 AR Marker

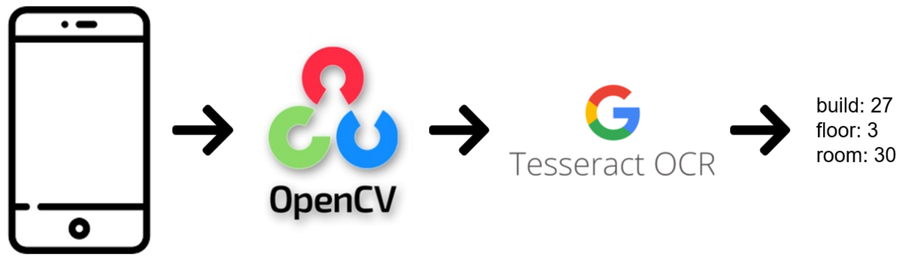

Figure 6: AR Marker Recognition Process

Indoor Positioning, a process of defining the user's indoor location. This is one of the most essential technical part of our service, for it is used to specify the user's current location, and correct errors that occurs during user's movement. For such this, a system called MARKER will be used.

The team will be using building numbers and classroom identification plates are to be used for MARKERS. To achieve this tesseract and OpenCV will be used to extract the number the camera is capturing.

tesseract is a program engine that returns an image containing text as one of the Optical Character Recognition(OCR) types. OCR is usually used for converting images to text in scanner apps. The tesseract engine we will be using is currently under development by Google, and its accuracy continues to increase through machine learning as the version increases.

OpenCV is a computer vision library for image processing. Even though the image taken from the camera is high quality, the data is not easily recognizable by the computer. Therefore, the team ill be implementing OpenCV as a form of a preprocessing tool for the computer to more easily recognize the data. With the processed data, through tesseract, the phone can recognize the building number and classroom number being capture on camera.

For both OpenCV and tesseract, the team will be using python language. Because, python has numpy library, which is related to OpenCV formula, and tesseract has pytesseract wrapped from python use.

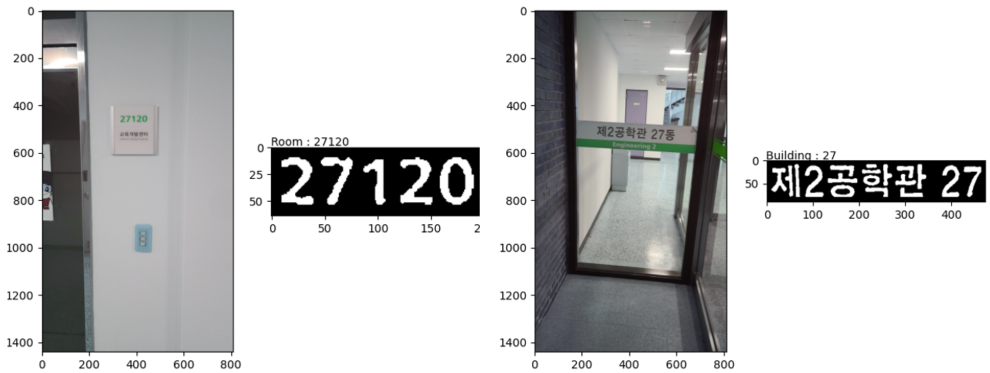

Figure 7: AR Marker Recognition Test

3.2.4 Anchoring

AR anchors leave virtual location information that specifies a location in the real world based on information obtained from real-world vision. This information is based on the detection of planes and feature points. A feature point is a characteristically distinct part of an image, which can be the edge of an object or a characteristic texture. In the case of planes, they can be thought of as grouped feature points that share the features of a planar surface. Algorithm to find feature points in images is a fairly old research field, and there are general algorithms to find feature points through blur of images, such as SIFT or SURF. However, blurring is not suitable, for it is too heavy to use in real time on a mobile device, resulting in performance degradation.

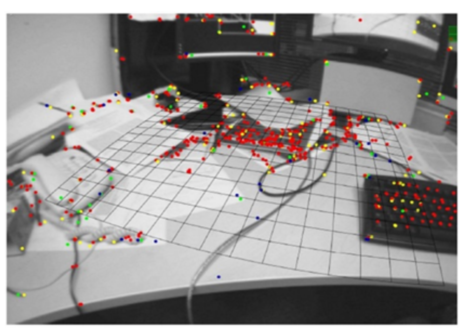

Figure 8: Feature Point Sample

Though the team would like to explain the algorithm of the SDK being used, Google and Apple did not disclose any information regarding the algorithm used to detect feature points in the AR core and AR kit. Therefore, this paper will briefly introduce one of the well-known feature point detection model, called BRISK algorithm.

BRISK algorithm analyzes the circular perimeter of each pixel in the image to determine whether a pixel's brightness is below or above a threshold. If a certain number of pixels meet this criterion, it is determined to be distinct from its periphery, classified as a feature point. For BRISK, at least 9 consecutive pixels in a 16-pixel circle must be at least a threshold brighter or darker than the center pixel.

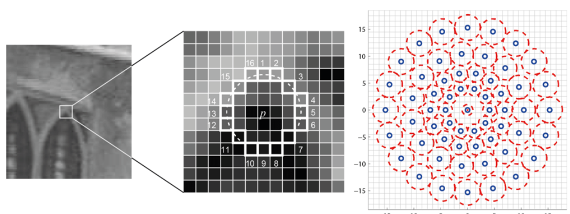

Figure 9: Feature Point Sample

Image recognized by the camera and the descriptor of the 3D mapping image stored in the database are compared via several samples surrounding the center feature point, such that the algorithm can even compare them at different angles and lighting. If the feature points are the same, as shown in the sample image below, they are displayed in green, and if they are different, they are displayed in red.

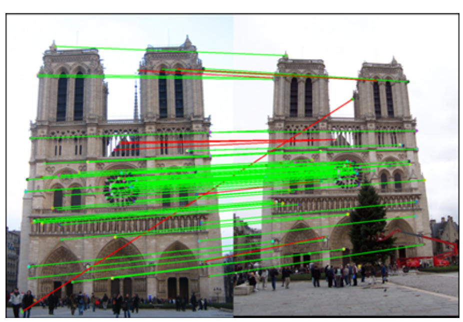

Figure 10: Feature Point Sample

The anchoring method we will use is a method of mapping feature points by scanning an object or the environment in 3 Dimensional space. An achor point stored in cloud is compared with feature points picked up in user's phone. When compared, the system can find out where the user is by finding the matching feature point pairs.

to obtain, the actual distance between the current device and the anchor can be inferred by triangulation as shown in the following formula, which improves the accuracy of the user's indoor positioning.

$$Z = \sqrt{h^2 + Y^2 + X^2} \quad (1)$$

$$X = Y \times \tan\varphi \quad (2)$$

$$Y = h \times \tan\theta \quad (3)$$

$$\varphi = \left(i - \frac{W}{2}\right) \times \left(\frac{FOV_H}{W}\right) \quad (4)$$

$$\theta = \omega + \left(\frac{H}{2} - j\right) \times \left(\frac{FOV_V}{H}\right) \quad (5)$$

Z: distance between camera and point

h: height of camera

φ: vertical angles

θ: rotation angles

Figure 11: Triangulation Formula

3.2.5 Navigation

As for navigation, if mapping of passages and spots is done well, it can be expressed as points and lines like an image, so it seems that you can set and guide routes based on graph algorithms such as A-star or Dijkstra. Since it is intended as a practical service, when there is enough time, the team will try to add functions to not only set simple destinations to class, but to also add in functions such as "finding nearby toilets, information desks, and entrances".

4 The Hurdles

The following are list of the team have discovered potential solution but did not pursue due to inadequacy or impracticality.

4.1 Global Positioning System

It is natural to consider Global Positioning System(GPS) for navigation. Upon research, When outside of the building without any hindrance, GPS had 5m 10m range of accuracy [?]. With such level of inaccuracy, it is inadequate for indoor navigation. Adding fuel to the fire, as the GPS's signal passes through concrete wall of the building, the signal goes very weak to the point where it has barely enough energy to be usable.

4.2 wifi

In addition, if we find a way to utilize wi-fi, we are considering using it together, or correcting through an algorithm based on this when a clearly distinct movement such as a user's rotation occurs at an intersection. However, due to the code's complexity and time limitation, it was not to be used.

4.3 Mapping

One form of Campus mapping the team have come up with was through visual image mapping data. Such method will require features to be extracted based on information input to visual sensors such as LiDAR or cameras. Robots with multiple sensors travel in space, as in photos, however, such method had to be let go due to limitation of capital.
Due to such reason, the team had to come up with a solution to develop an indoor navigational system where the program could run using only extremely limited cloud server and user's CPU in the phone.

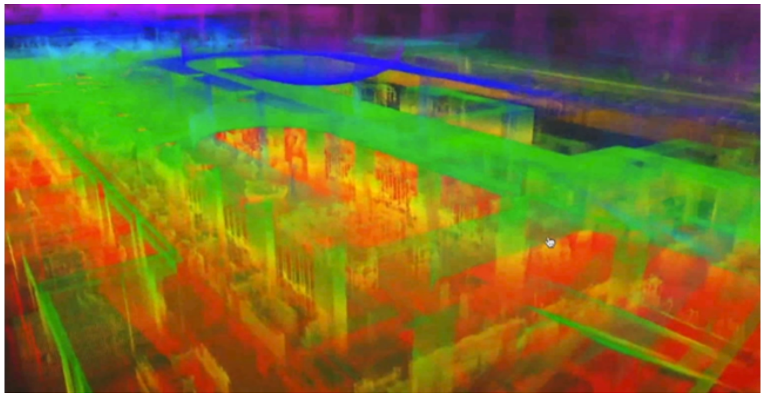

Figure 12: lidar

5 Planning in Detail

Stage || Time Period	w1	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12
Design & Work Division												
Pseudo Code Development												
Front-end build												
AR script build (Space)												
AR script build (Nav)												
Bug Testing												

Figure 13: Team Development schedule

5.1 Division of Labor

5.2 1~2 Weeks

In the 1st and 2nd weeks, since there is still no solution that can be said to be the correct answer for the implementation of AR indoor navigation, the team is will be collecting and analyzing related previous studies as much as possible to select candidates suitable for the scale we will try, and Implement AR experiences

AR Development & Mapping	Jun-seo Mah
AR marker recognition	Jung-Min Ahn, Min-gu Cho
Vision based localization	Hyon-ho Kim, Jung-Min Ahn
Location error correction	Hyon-ho Kim
Navigation algorithm	Min-gu Cho
Data management	Min-gu Cho
Create app flow, UI/UX	Jun-seo Mah

5.3 3~4 Weeks

In the 3rd and 4th weeks, the selected technologies are implemented so that they can be tested on a small scale of about two corridors. During this week, the mapping and localization technologies that are considered suitable are selected.

5.4 5~9 Weeks

Between the week of 5th and 9th, the entire system such as mapping, localization, and route setting algorithm is implemented and tested at the building scale. For the mapping, the team will be getting school schematics and build it in ”BENZENE-ring” building. For Localization, the team will be taking several photos of the building for feature points and Class room numbers. Though it hasn’t not been finalized, the triangulation algorithm will be implemented.

5.5 10~Weeks

After the 10th week, the project is integrated and finalized. If there is room in the schedule, the system could correct localization errors through algorithm and improve accuracy. The team will conduct further research to improve UI/UX in AR environment, or provide service in a more general environment, expanding service beyond the ”Benzene-ring” building.