

드론 신속 배송을 위한

# AI 기반 최적 하늘길 구축 시스템

하늘길을 활용한 병원 응급 물품 배송



'달리는 차 위에 나는 드론 있다'

김기현 박광민 조남현

# | 목차



# 1. 프로젝트 개요 주제 선정 배경

## 골든타임 사수의 필요성



혈액원에서 한 시간 이상 거리 위치 병원,  
수혈 지체 시 응급환자 사망률 **99.9%**

## 혼잡한 도로의 배송지연



스웨덴 구조 실험서 의료용품 매단 드론이  
앰불런스보다 **17분 빨리** 도착

## 드론 배송의 성공사례



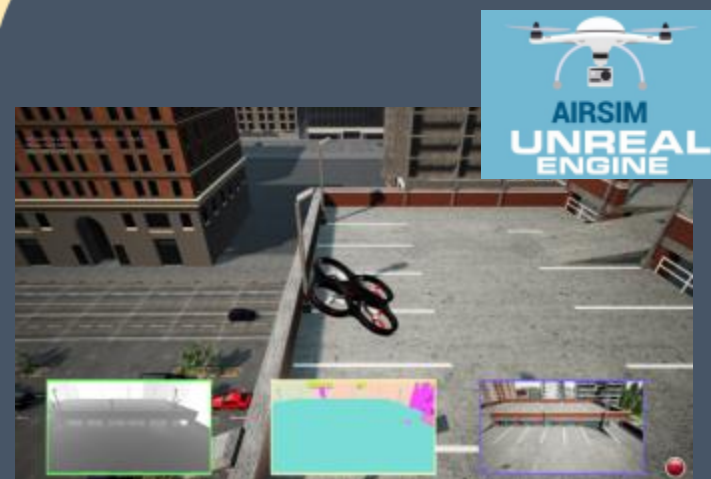
르완다, 드론으로 수혈용 혈액 수송  
소요시간 **4시간에서 15분**으로 단축

# 1. 프로젝트 개요 주제 선정

“ 드론 응급물품 배송 을 위한  
한국 지형 특성 반영 하늘길 구축 ”



한국 : 다수의 고층 건물과 산, 도심 인구 밀집



가상환경 시뮬레이션

특성 반영



한국 지형 반영 최적 하늘길 도출



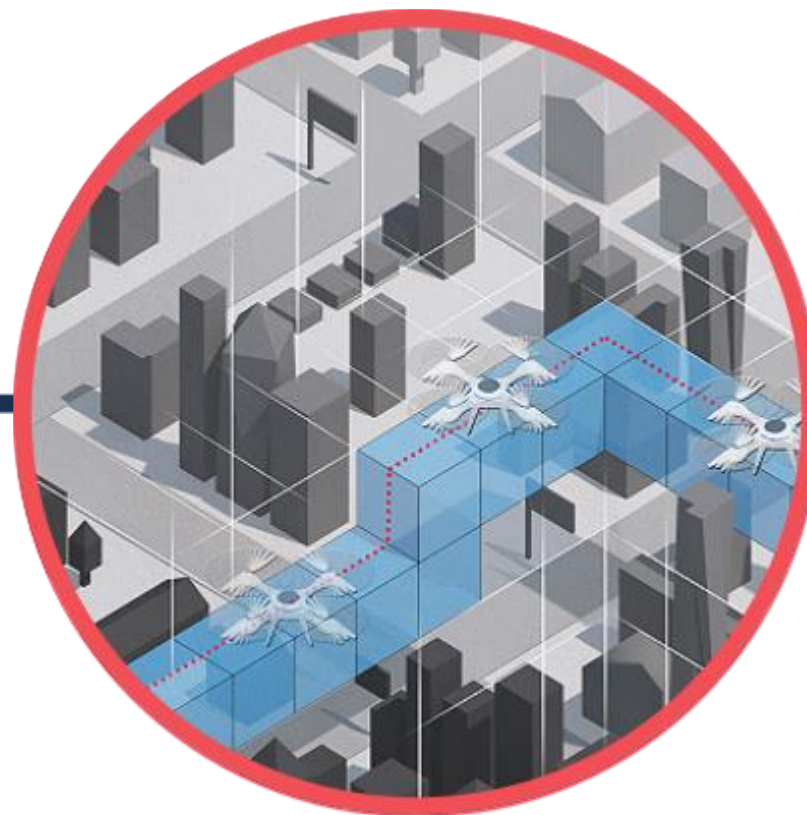
# 1. 프로젝트 개요 활용 시나리오

## 1. 긴급 상황 요청 발생



긴급상황 혈액 요청이 발생하면 사용자는  
출발지와 도착지를 설정

## 2. AI 기반 최적 하늘길 계산



AI 기반 최적 하늘길 계산은 날씨와 건물 등을 반영  
하는 비용함수를 통해 이뤄짐

## 3. 혈액원 내비게이션 활용



최적 경로를 화면에 띄워줌으로써  
내비게이션으로 활용 가능

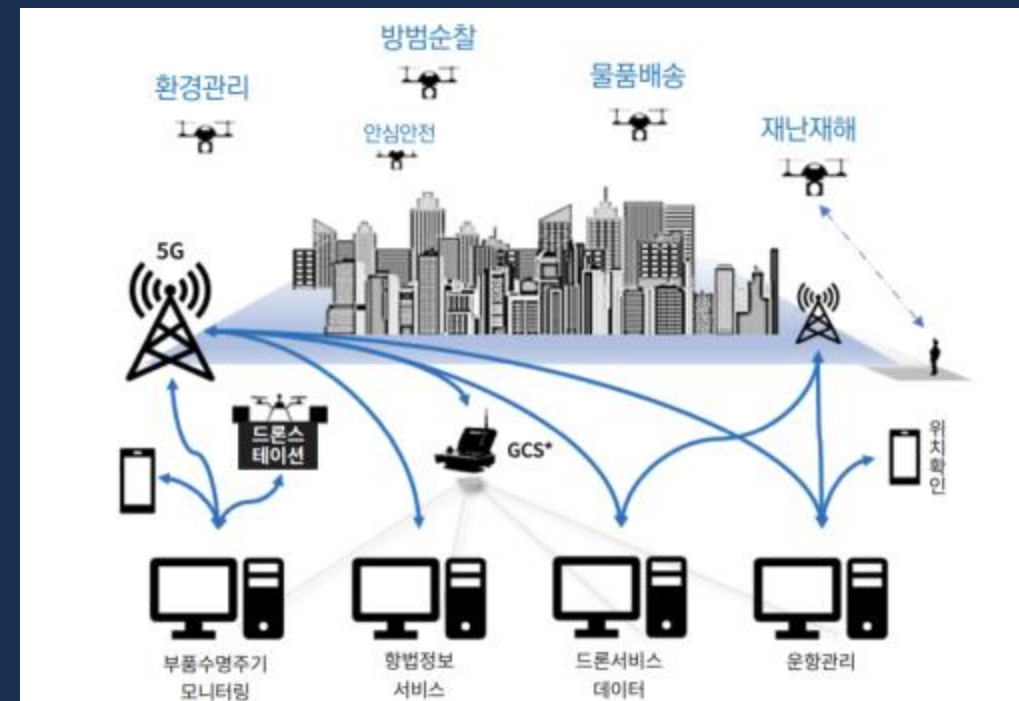
# 1. 프로젝트 개요 기대가치

## 사회적 비용 절감



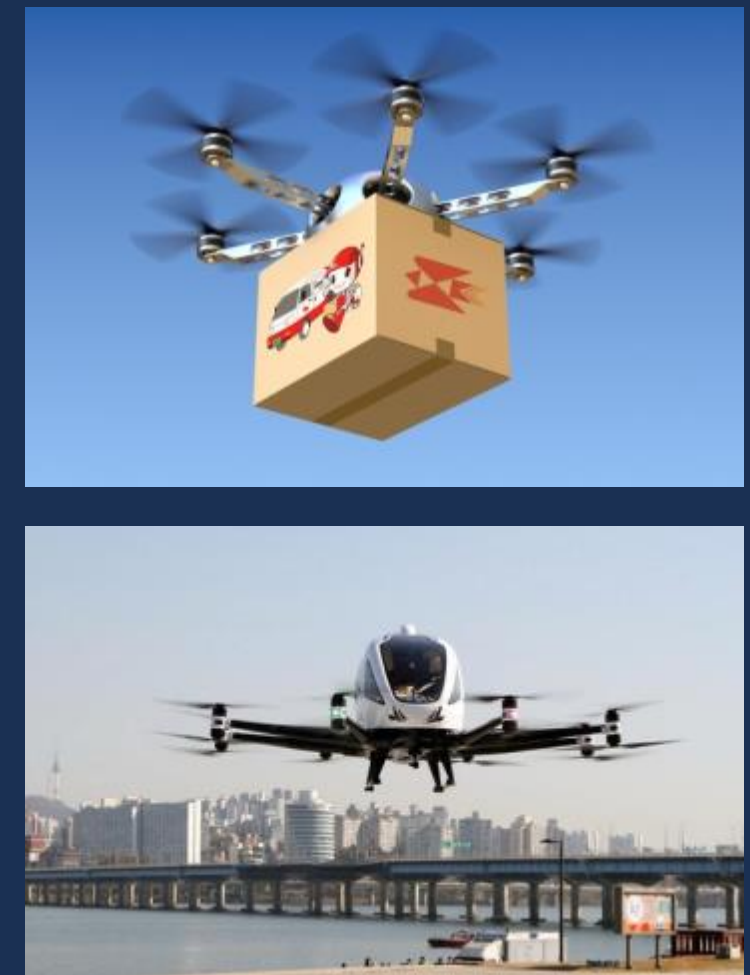
도로 운송 대비 평균 80%의 시간 절약

## 데이터 창출



실제와 유사한 드론 비행 데이터로  
드론 개발, 인프라 구축 선행 연구에 도움

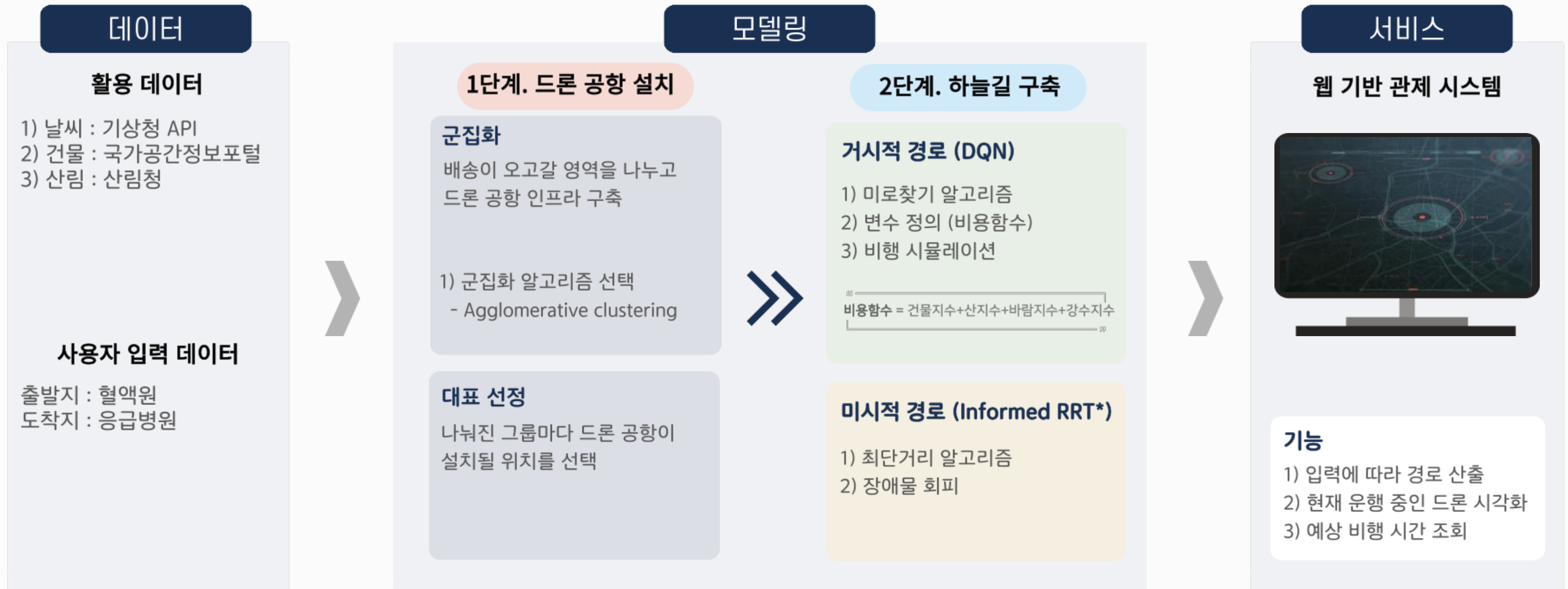
## UAM 확대 적용



드론 택배, 드론 택시 등 경로 선택 활용



## 2. 프로젝트 환경 시스템 구성



## 2. 프로젝트 환경 개발환경 & 활용 데이터

### 개발환경

사용 언어	python™ JS HTML
개발 환경	colab Google Cloud Platform MariaDB Foundation Visual Studio Code
라이브러리	TensorFlow GeoPandas
지리 정보	kepler.gl QGIS
시뮬레이션	AIRSIM UNREAL ENGINE
웹 구현	django

### <활용 데이터 및 데이터 베이스 구축>

기상 데이터	산림 데이터	응급병원 데이터	건물 데이터
 기상청	 산림청	 보건복지부	 국토교통부
강수량 풍향 / 풍속	폴리곤 좌표 정보	병원 좌표 정보 병원 기본 정보	폴리곤 좌표 정보 높이 정보



데이터 베이스 구축을 통해  
새로운 정보 업데이트 시 즉각 대응 가능



### | 3. 시스템 개발 개발 단계

#### 01 드론 공항 설치

#### 드론 배송 인프라 구축

하늘길 구축 전 드론 공항(혈액원)을 설치합니다.  
원활한 혈액 배송을 위한 혈액원 구축이 선행됩니다.

#### 02 하늘길 구축

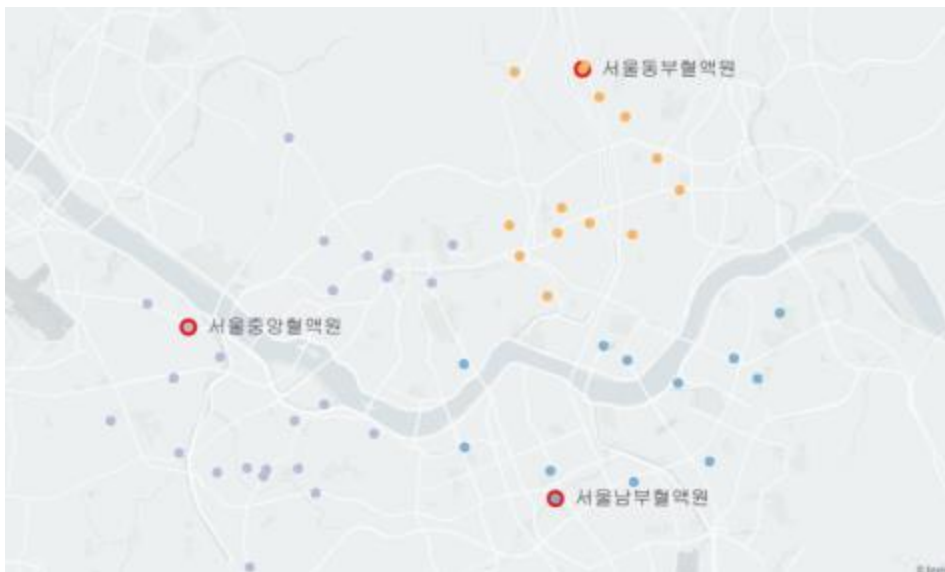
#### 최적 경로를 위한 2단계 거리 계산

환경 요소를 고려한 최적 경로는 거시적 경로 산출과  
미시적 경로 산출 두 단계를 통해 계산됩니다.



### 3. 시스템 개발 1단계. 드론 공항 설치

#### 드론 공항 설치의 필요성

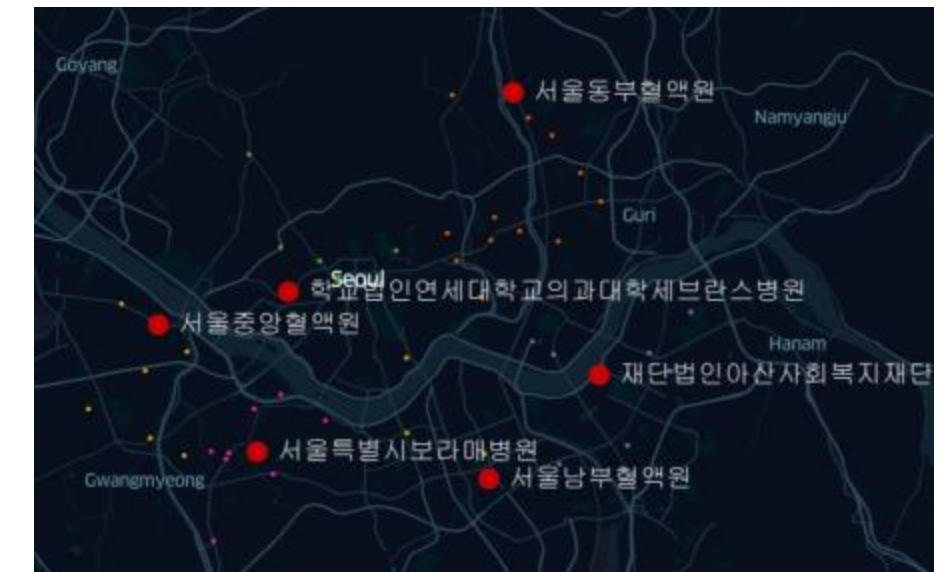


현재 서울은 단 3개의 혈액원 보유,  
하나의 혈액원이 담당하는 구역 방대

#### 군집화를 통한 임시 혈액원이 될 병원 선정

계층적 군집 (Hierarchical Clustering)	분할적 군집 (Partitional Clustering)

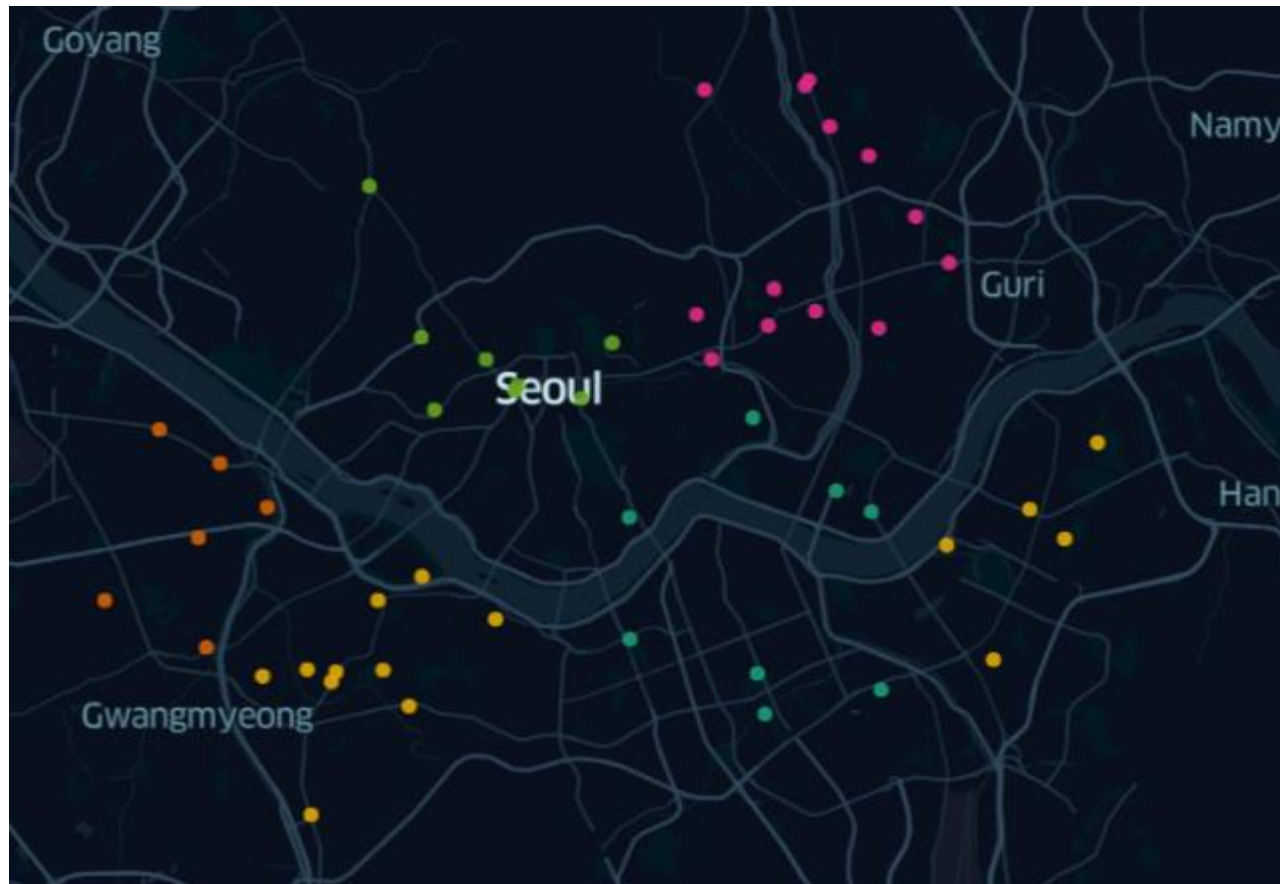
1. 군집화 알고리즘 선정



2. 군집화 기반으로 대표 병원 선정

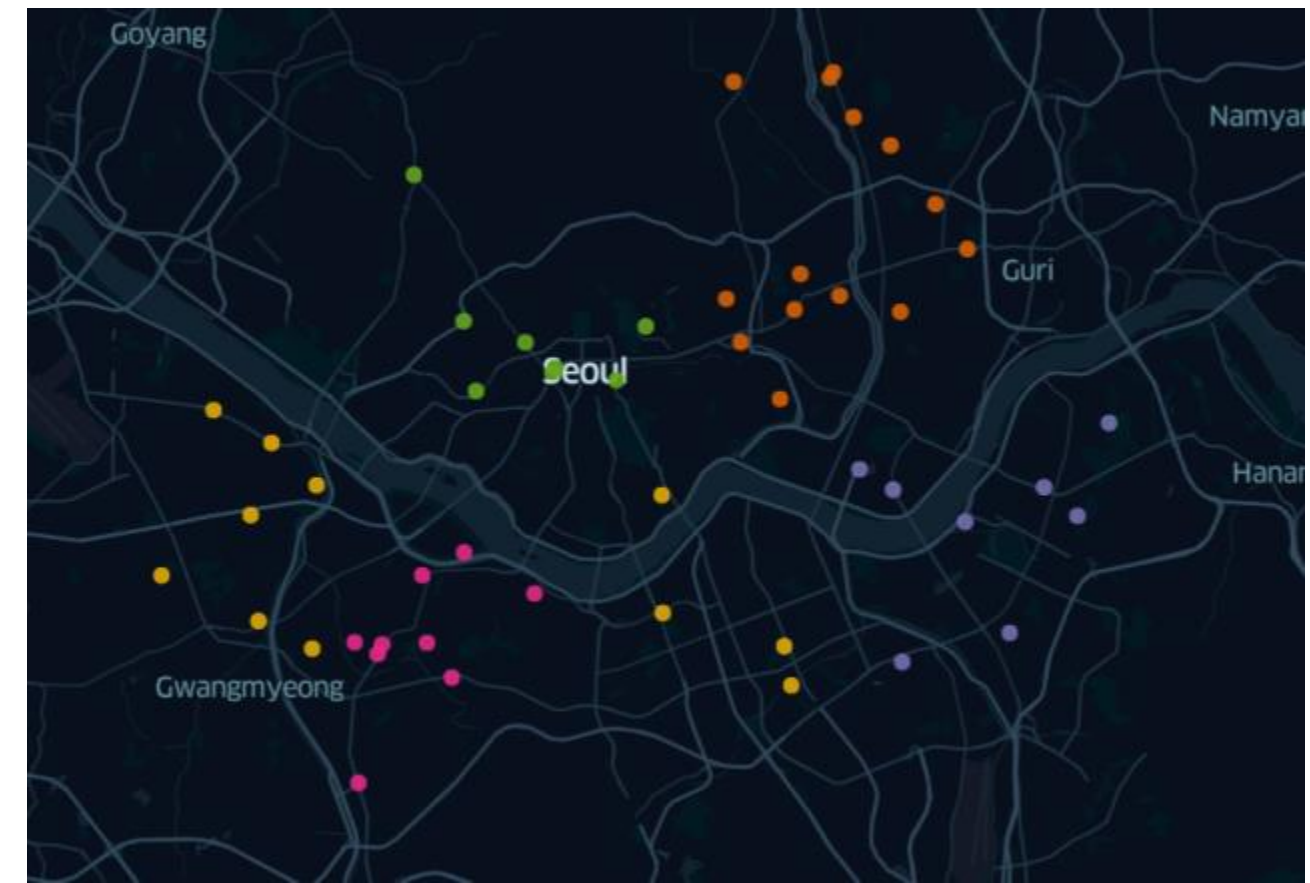
### 3. 시스템 개발 1단계. 드론 공항 설치 - 군집화

#### 1) 군집화 알고리즘 선정



K-평균 군집화

- 이상치에 취약
- 군집 중심을 랜덤으로 정하기 때문에 실행마다 군집이 변함
- **자료가 복잡해지면 분류가 잘 되지 않음**



응집형 계층적 군집화

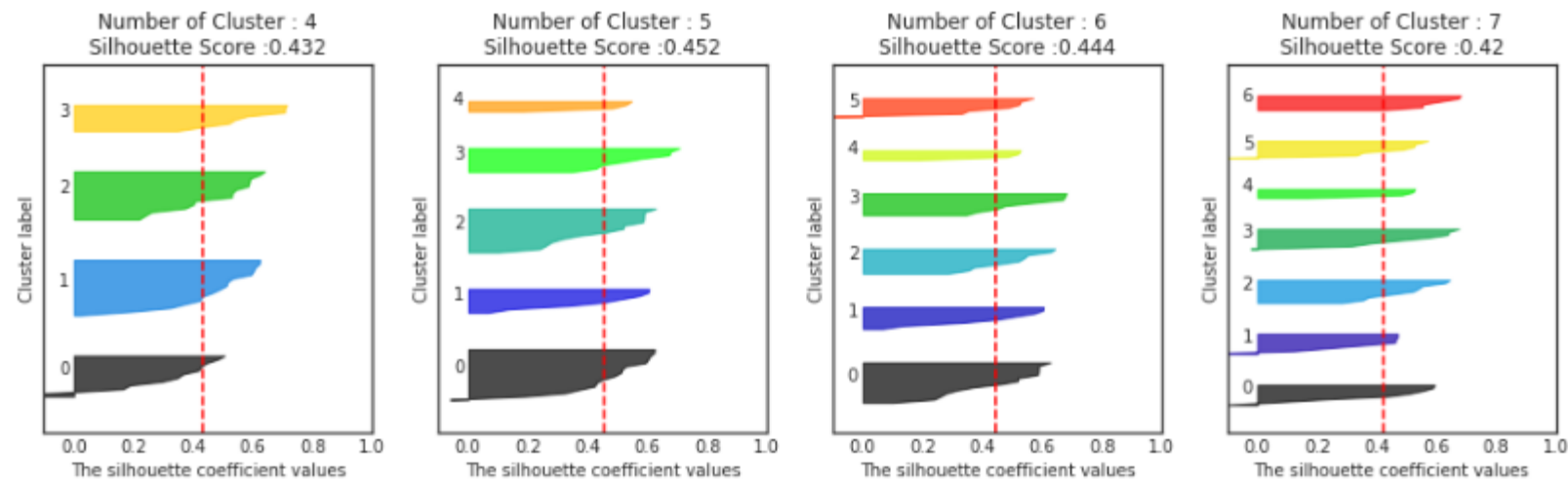
- 이상치에 강함
- 비슷한 크기의 군집으로 묶어줌

→ 추후 UAM으로의 확대 시 데이터가 복잡해질 가능성이 있으므로 이것을 선택



### 3. 시스템 개발 1단계. 드론 공항 설치 - 군집화

#### 2) 군집화 기반으로 대표 병원 선정



- 실루엣 계수는 군집개수 5개와 6개가 비슷
- 군집 내 병원 간 **최대 거리가 10km 이내인 6개**로 결정



1. 응급실, 입원실, 수술실, 병상수 ↑

- 충분한 혈액 수용 시설
- 혈액 수요가 높아 우선공급에 이점

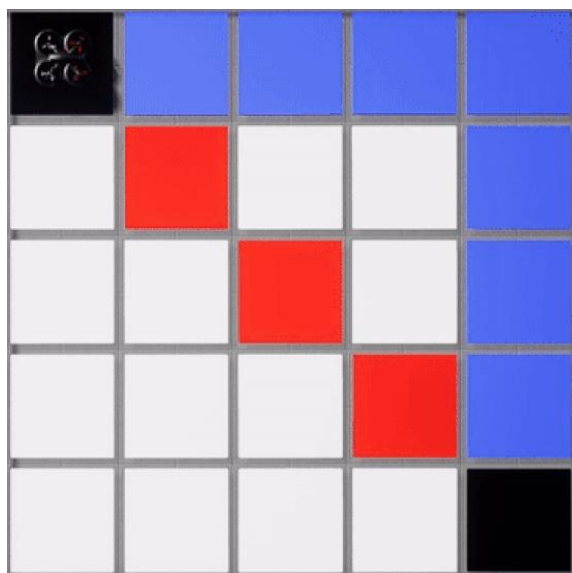
2. 군집 중심까지의 거리 ↓

- 군집 내 배송에 이점

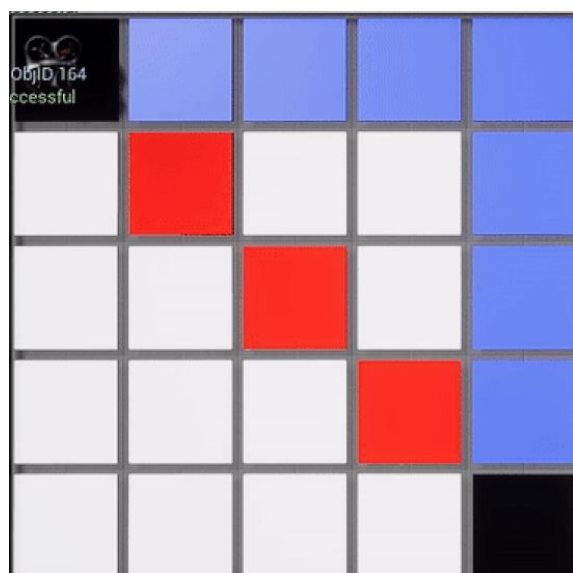
### 3. 시스템 개발 2단계. 하늘길 구축

최적 경로 = 거시적 경로 + 미시적 경로

< 최단 경로 >



< 최적 경로 >



최단 경로 ≠ 최적 경로

⇒ 배터리 상태와 환경 요소 등을 고려한 최적 경로의 산출 필요

거시적 경로 산출

- 1) 미로찾기 알고리즘 (DQN)
- 2) 변수 정의 (비용함수)
- 3) 비행 시뮬레이션

미시적 경로 산출

- 1) 최단거리 알고리즘 (Informed RRT\*)

### 3. 시스템 개발 2단계. 하늘길 구축 - 거시적 경로 산출

#### 1) 미로찾기 알고리즘 (DQN)

##### < 강화학습 방법 >



<미로 정보 스캔>

Start



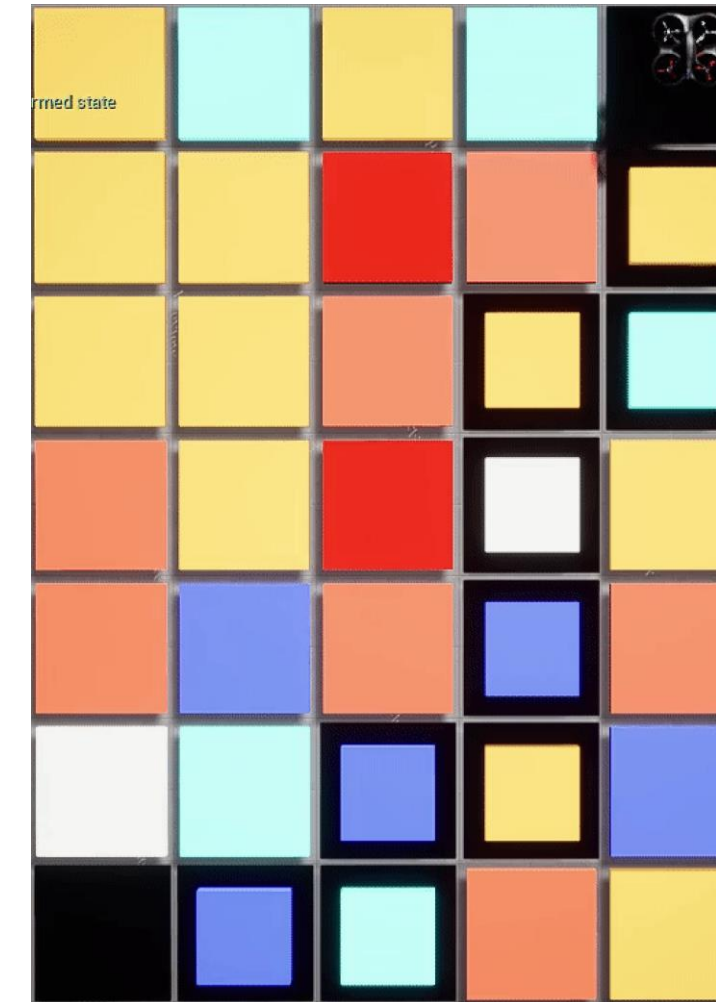
-500

Goal  
+ 500

DQN (Deep Q-Network) :

- 구글 딥마인드사에서 개발한 심층강화학습
- Q-learning 강화학습 알고리즘에 심층신경망을 적용
- 특정 상태에서 특정 행동을 취할 때 가장 큰 보상을 받는 방법으로 학습

##### < DQN으로 산출한 경로 >



Reward 감소

최적 경로 산출 :

- 실제 공간을 1km 범위로 격자화
- 드론 비행에 영향을 주는 환경 요소를 비용함수화
- 비용함수를 보상으로 적용하여 학습 후 경로 산출



### 3. 시스템 개발 2단계. 하늘길 구축 - 거시적 경로 산출

#### 2) 변수 정의 - 드론 비행에 영향을 주는 환경 요소



##### 건물 지수

- 좌우 회피 기동은 건물 수에 비례
- 좌우 회피 기동이 많아질수록 배터리 소모 증가 및 비행 시간 증가
- 격자 안의 건물 개수를 기준으로 지수화



##### 산 지수

- 상하 회피 기동은 산 면적 만큼 지속
- 상하 회피 기동이 지속될수록 배터리 소모 증가 및 비행 시간 증가
- 격자 안의 산 면적을 기준으로 지수화



##### 바람 지수

- 역풍 속력에 비례하여 추가 동력 필요
- 추가 동력 만큼 배터리 소모 증가
- 격자 안의 풍향, 풍속을 기준으로 지수화
- 순풍은 역풍과 반대 효과
  - 역풍 reward : 패널티
  - 순풍 reward : 보너스



##### 강수 지수

- 강수량에 비례하여 시야 제어 영향 및 고장 발생
- 격자 안의 시간당 강수량을 기준으로 지수화

$$\text{비용함수} = \text{건물 지수} + \text{산 지수} + \text{바람 지수} + \text{강수 지수}$$

### 3. 시스템 개발 2단계. 하늘길 구축 - 거시적 경로 산출

#### 3) 비행 시뮬레이션 - 변수 회귀식 도출

시뮬레이션 환경

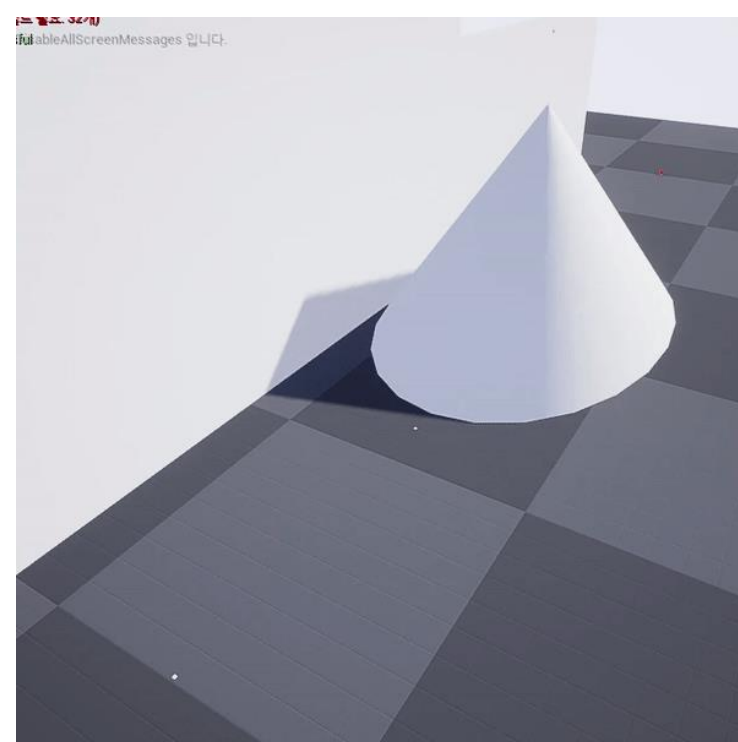
길이 : 50m, 폭 : 20m, 고도 : 20m, 시속 : 70km/h



< 건물 >

독립변수 : 건물 수  
종속변수 : 시간

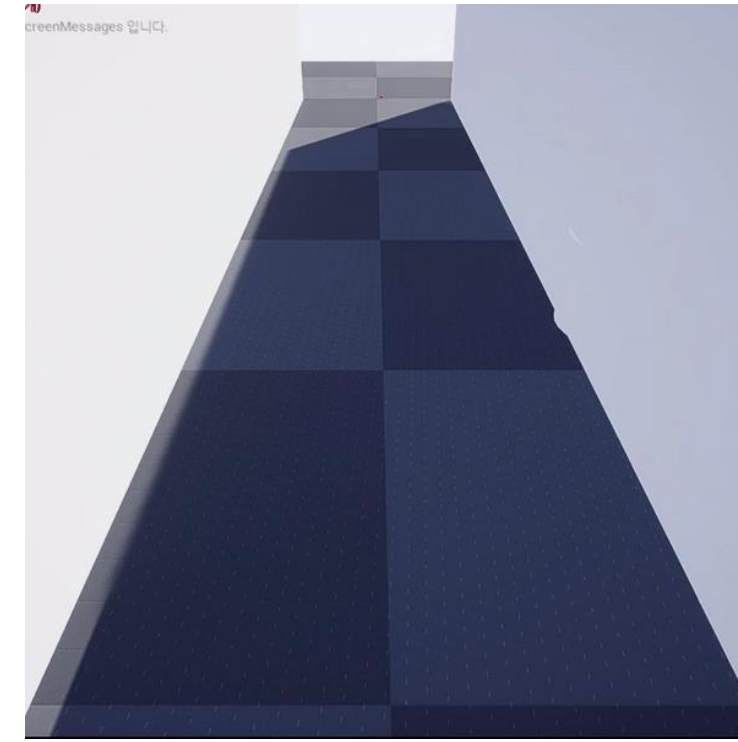
회귀식 :  
 $y = 0.8086x + 49.5$



< 산 >

독립변수 : 산 면적  
종속변수 : 시간

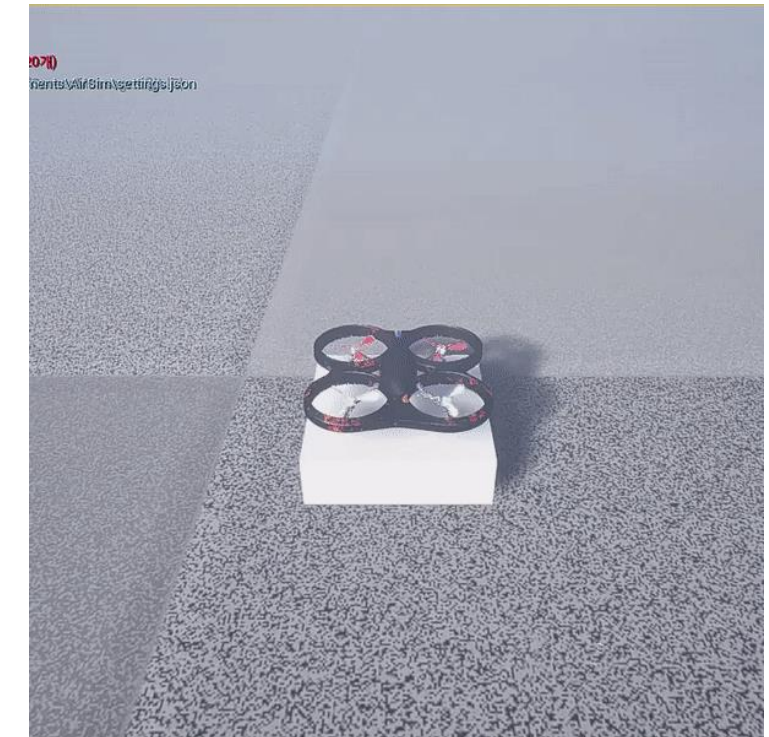
회귀식 :  
 $y = 0.7253x + 39.67$



< 바람 >

독립변수 : 풍속  
종속변수 : 시간

회귀식 :  
 $y = -0.828x + 51.91$



< 강수 >

시간 지연은 확인 불가  
시야 좁아져 객체탐지에 영향  
고장의 원인

-> 기상청 강수량 등급 사용

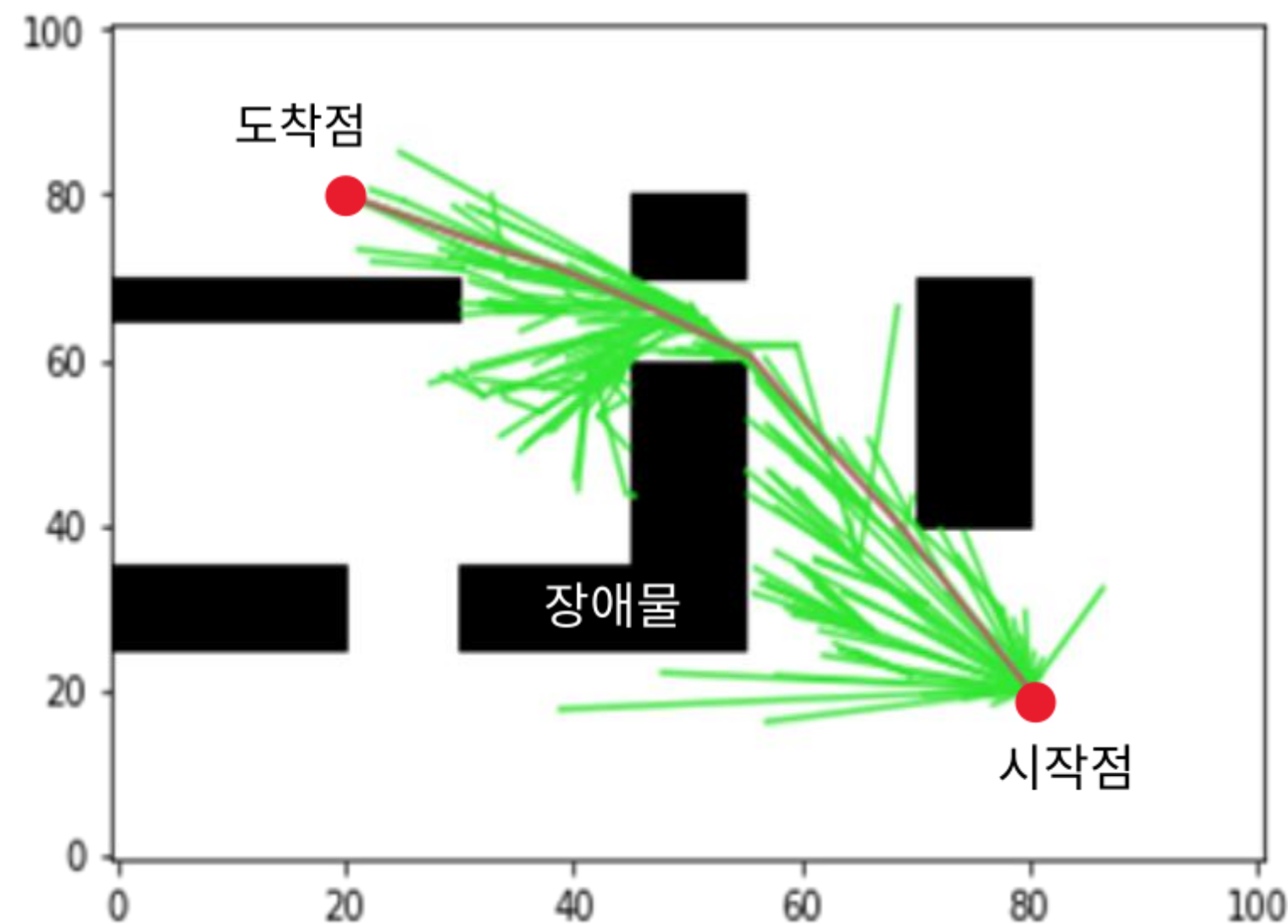
### 3. 시스템 개발 2단계. 하늘길 구축 - 미시적 경로 산출

Informed RRT\* :

- 샘플링 기반 최단 거리 알고리즘

장애물 회피 :

- 거시적 경로를 활용하여 지도를 골격화
- 골격화된 경로 안에서 건물 등의 장애물을 회피하여 최단 경로 산출



< Informed RRT\*로 산출한 경로 >



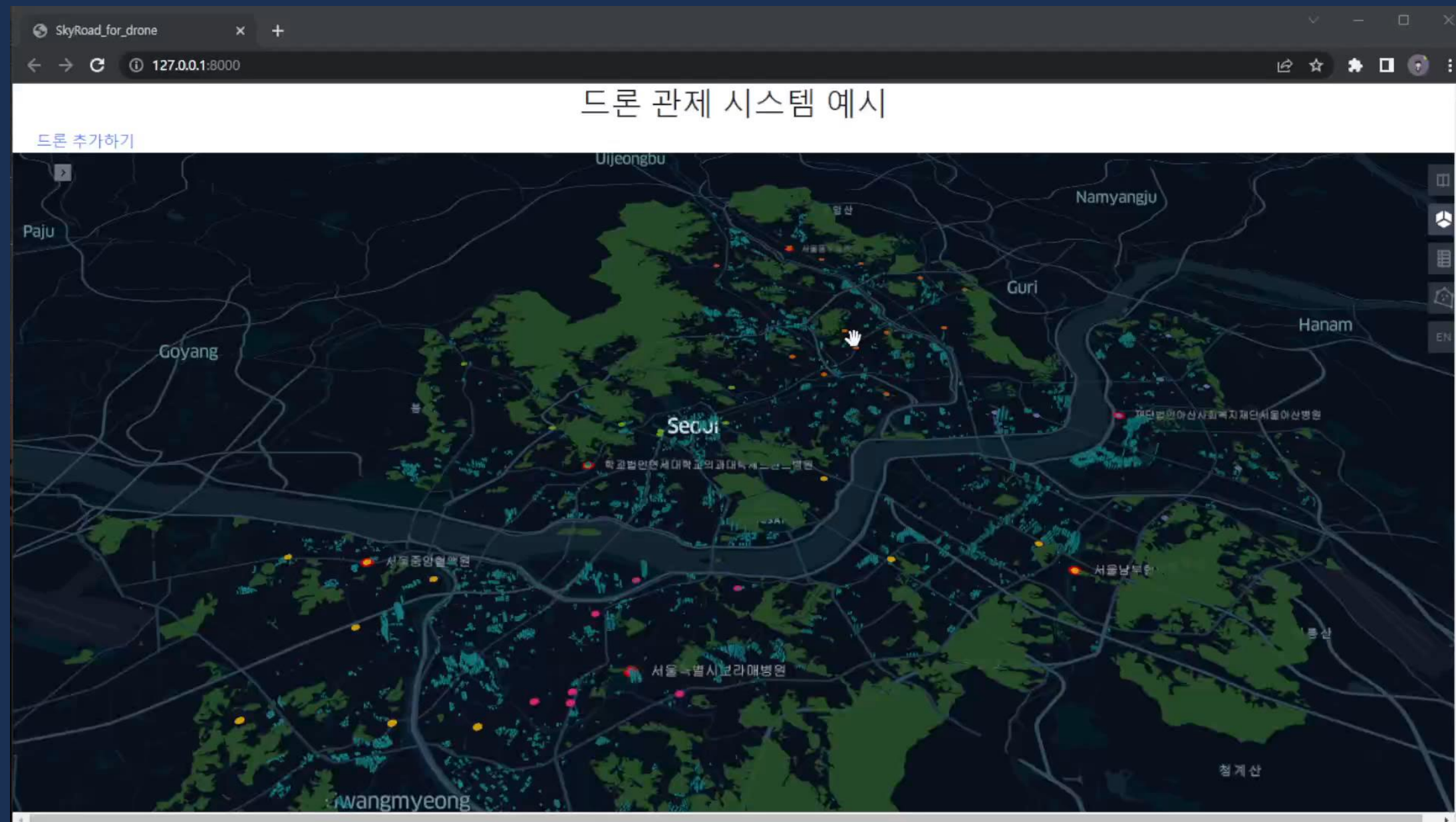
## 4. 시스템 시연 최종 서비스 웹 구현

### 배송 드론 스펙

평균비행속도 : 70km/h   최고비행속도 : 105km/h   비행고도 : 50~150 (m)   드론무게 : 4.5kg   최대적재중량 : 4.5kg   이동가능거리 : 왕복 35km



### 웹 구현



## 4.5. 코드 설명

### 군집화, 최적 경로 산출, 웹 구현

The screenshot shows a Jupyter Notebook environment in Visual Studio Code. The workspace contains several files, including '드론공항설치.ipynb', '최적경로산출.ipynb', and '경로 시각화 (경로추가포함).ipynb'. The active notebook is '드론공항설치.ipynb', which is open to a cell titled '데이터 준비' (Data Preparation). The cell contains Python code for setting up a database connection and updating data.

**드론공항설치.ipynb**

Project\_3 > sourcecode > 드론공항설치.ipynb > 데이터 준비 > # db에 업데이트하는 사용자 함수

Successfully installed pymysql-1.0.2

### 모듈 임포트

1 cell hidden ...

### 데이터 준비

```

1 # db에 업데이트하는 사용자 함수
2 def update_db(df, df_name):
3     import pymysql
4     from sqlalchemy import create_engine
5
6     db_connection_str = 'mysql+pymysql://root:ASewetsvc124~1242#%1wsdehXV@34.64.132.212:3306/source'
7     db_connection = create_engine(db_connection_str)
8     conn = db_connection.connect()
9
10    df.to_sql(name = f'{df_name}', con = db_connection, index = False, if_exists = 'replace')

```

[3] Python

### 응급병원 위치 데이터

- 공공데이터포털 API - 중앙의료원\_응급의료정보

4 cells hidden ...

### 응급 의료기관 기본 정보 데이터

2 3

Jupyter Server: local Cell 5 of 43



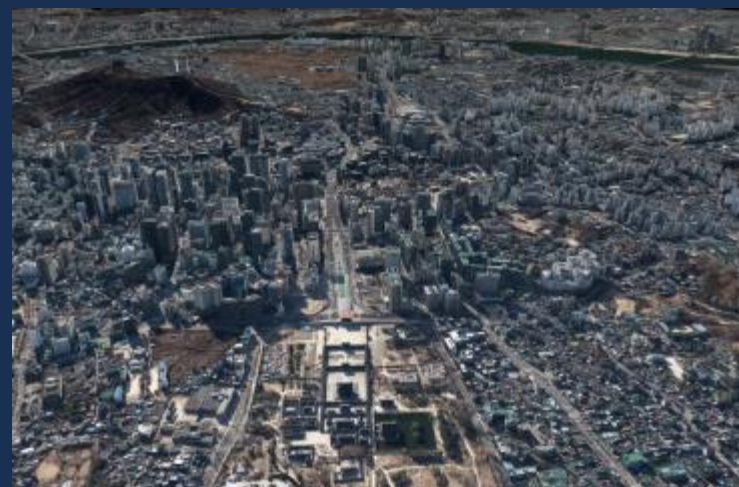
## 5. 발전 가능성 시스템 개선 방향

### 모델 개선



격자 세밀화, 파라미터 튜닝 등  
모델 최적화를 통한 속도 향상, 일반화

### 서울 3D 지도 데이터 활용



S-Map 데이터에 외부요인 추가 후  
실제와 유사한 시뮬레이션 구현

### 드론 전문가 협업



드론 전문가와 협업을 통한 추가 변수  
개발 및 비용함수식 개선

### 사회적 이슈 반영



사생활 침해 우려, 소음문제 등의  
사회적 이슈가 될 부분 개선



# 소감

## Team 광기남

<https://github.com/CAKD5ASSEMBLE>



### 김기/현 팀장

사실 프로젝트 초기에는 셋이서 결과물을 내기 벅찬 주제가 아닌가 싶었지만 팀원 모두가 유능해서 그런지 잘 마무리 지을 수 있었다. 이번 결과물은 나름대로 만족했다. 다만 한가지, 서울의 3D 공간 데이터를 사용하려면 너무 비싼 비용을 지불해야해서 서울을 구현하여 시뮬레이션하지 못한 점이 아쉽다.

코드의 구조는 보다보면 이해할 수 있지만 코드의 근본이 되는 공식과 수식을 완벽히 이해하지 못하고 사용하는 것이 항상 마음에 걸린다. 이 세상 모든 원리는 수학을 떼 놓고 생각할 수 없는데 이 부분에서 아직도 갈 길이 멀구나 싶다.

GitHub

<https://github.com/spiner321>

### 박광민 수석 프로그래머

세 명이 같이 하였기에 현재의 결과가 나왔다고 생각합니다. 이번 프로젝트에서 새로운 툴과 알고리즘을 사용하였고 의미있는 결과가 나와 기쁩니다.

몇 가지 아쉬운 점은 이상치가 많은 데이터의 가공을 충분히 하지 못하였고, 좀 더 완벽한 시뮬레이션이 이루어지지 못한 것에 많은 아쉬움을 느낍니다.



GitHub

<https://github.com/KMP94>

### 조남현 수석 프로그래머

마음이 잘 맞는 팀원들과 함께 하니 적은 인원임에도 결과물을 낼 수 있었던 것 같습니다.

구현하고 싶은 것들에 비해 얻을 수 있는 데이터가 한정적이어서 아쉬웠지만, 다양한 언어와 요소기술들을 활용하면 어느 정도 극복이 가능함을 깨닫는 유의미한 시간이 되었습니다.



GitHub

<https://github.com/MiddleJo>