

2023 KETI 고려대 AI 융합·응용기술 아이디어 경진대회

AI 학습기반 산사태 예측 및 실시간 대응 시스템 아이디어

고려대학교 통계학과

원윤정 (jennie0715@korea.ac.kr)
김혜연 (yeorii@korea.ac.kr)
장희중 (vicky4150@korea.ac.kr)



INDEX

1

분석 개요

- 분석 주제 및 목적
- 분석 순서도

2

데이터 수집 및 전처리

- 데이터 수집
- 데이터 전처리
- 변수 설명

3

산사태 예측 모델링

- Segmentation
- 모델링 & 튜닝
- 최적 모델 선정
- 변수 중요도 해석

4

실시간 대응 시스템

- 분석 결과 시각화
- MSC 도식화
- MSC 활용방안

5

결론

- 기대효과
- 의의 및 발전 방향

6

툴·데이터 및 참고문헌

7

부록

Part 1

분석 개요



분석 주제 및 목적

AI 학습기반 산사태 예측 및 실시간 대응 시스템 아이디어

1



도로변 산사태를 감지하여
이를 운전자와 산림 관계자에
알림

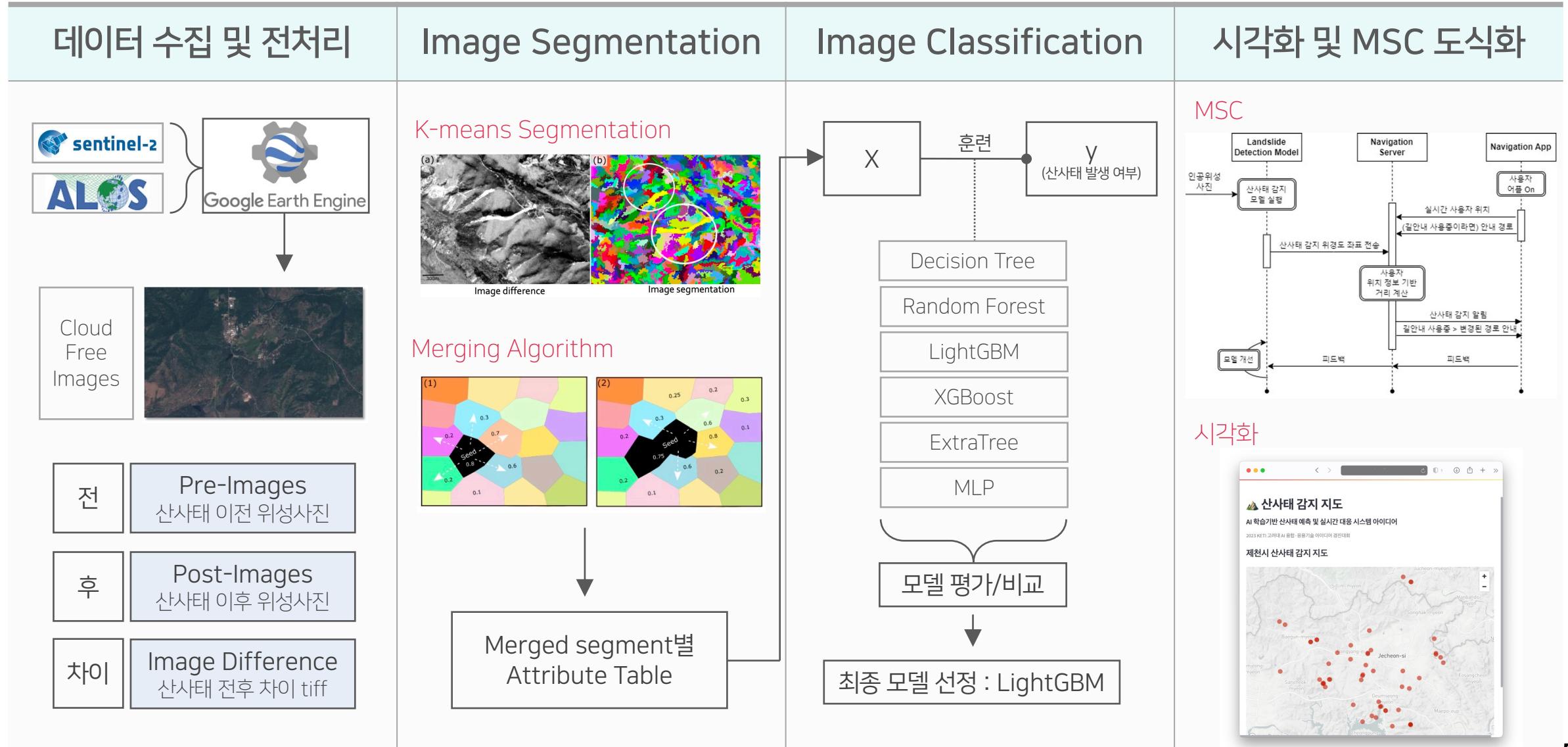
2



CCTV 데이터 수집 불가
→ 구글 어스에서 무료로 제공하는
인공위성 이미지를 활용하여
산사태 감지 모델을 개발

분석 순서도

산사태 감지 모델링



Part 2

데이터 수집 및 전처리



데이터 수집

산사태 발생이력 데이터 수집 - 제천시

- 포함 변수: 발생일자, 주소, 위도, 경도, 피해구역, 피해면적(ha)
- 총 산사태 발생 건수 74건

재해명	발생일자	종료일자	피해구분	국_사구분	주소	Latitude	Longitude	피해구역	피해면적(ha)
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시고암동102-1	37.1665011	128.2431537	제천시 고암동 102-1번지 외 4개소	1.04
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면구룡리산31-1	37.0590931	128.165182	제천시 금성면 구룡리 산31-1번지 외 3개소	0.73
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면대장리산42	37.0634327	128.1843958	제천시 금성면 월굴리 산32번지 외 3필지	0.4
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면동막리산119	37.1063793	128.2150441	제천시 금성면 동막리 산119 외 4필지	0.48
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면사곡리산35-3	37.0795983	128.1468517	제천시 금성면 사곡리 산35-3번지 외 3개소	1.4
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면양화리산87	37.0917307	128.2052785	제천시 금성면 양화리 산87 외 3개소	0.38
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면양화리산8	37.0680458	128.1961131	제천시 금성면 대장리 산42번지 외 1필지	0.46
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면양화리산15	37.0702037	128.2036203	제천시 금성면 양화리 산15 외 3개소	0.5
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면양화리산67-1	37.0811978	128.2063055	제천시 금성면 양화리 산67-1 외 4필지	1.01
집중호우	2020-07-28	2020-08-11	산사태	사유림	충청북도제천시금성면양화리산40-14	37.0755147	128.2175595	제천시 금성면 양화리 산40-14 외 1개소	1.1

데이터 전처리

Google Earth Engine (GEE)

Sentinel-2



1. Cloud Masking
산사태 감지를 위해 이미지에서 구름을 걷어냄
2. 픽셀 단위로 features 계산
 - NDVI, Brightness, Slope_max, Slope_mean, Height_max, Height_min 등
3. Image Difference (Pre-image vs. Post-image)
 - 산사태 발생 전후의 이미지의 차이를 feature로 추출
 - Vegetation index Difference, RG_Difference, Brightness_Difference 등

변수 설명

NDVI (Normalized Difference Vegetation Index)

상대적 생물량이라고도 알려진 녹색이 표시된 이미지를 생성할 수 있는 표준화된 지수

$$ndvi(i, j) = \frac{gq(i, j) - gk(i, j)}{gq(i, j) + gk(i, j)} \quad (i, j) : i = 1, \dots, n; j = 1, \dots, m$$

RGD (Red/Green Difference)

Red/green ratio의 차이

$$rgd(i, j) = rg(i, j)_{t_2} - rg(i, j)_{t_1} + c \quad (i, j) : i = 1, \dots, n; j = 1, \dots, m$$

Part 3

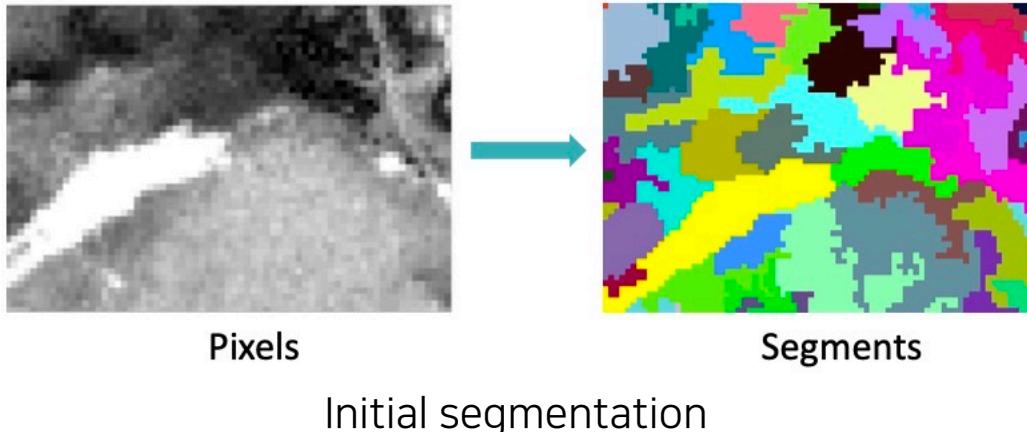
산사태 예측 모델링



Segmentation – Initial segmentation

- Red/Green Difference (RGD)를 feature로 하여 pixel 단위를 segment 단위로 바꾸어, 이후 segment based 분석 진행
- 산사태 발생 위/경도로부터 피해 면적(ha) 크기의 버퍼를 생성하여, 해당 버퍼에 포함된 segment에 class=1(산사태 발생) 할당

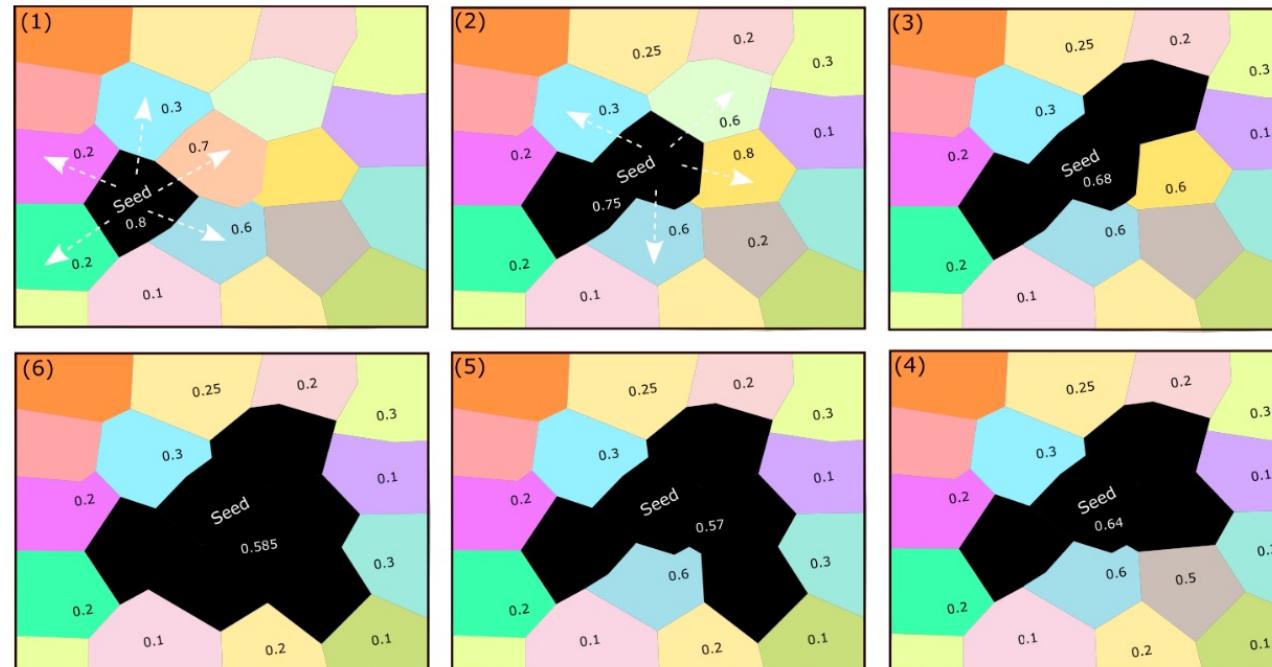
✓ 공간 정보, 산사태 발생 여부(class) 정보, NDVI, brightness, slope, height 정보들이 각 segment에 저장됨



segment_id	class	ndvi	ndvi_change	ratio_rg_change	brightness
L1_S1180	1	0.25832504927803300	-0.05560182806606740	0.44075197722371800	1.325204420814200
L1_S1208	1	0.28063626766204800	0.047349554086104000	0.6626704637209570	1.1643200063705400
L1_S1245	1	0.3352930048430290	-0.14736842174661900	0.3239077455490020	1.1099504514320500
L1_S1268	1	0.29591907854598200	0.00709864220190512	0.4446761348103520	1.0608008955949100
L1_S1	0	0.6307297103545250	-0.05310715118055270	0.4586822272849680	0.7173529432601290
L1_S2	0	0.626831517724568	0.009263465905864840	0.48420965040258600	0.7096141197998540
L1_S3	0	0.6279230346479040	-0.03426156531213110	0.46750678881855300	0.7211683140237730
L1_S4	0	0.6071622106483430	0.053180581562505200	0.5200168442423060	0.7417711858022010
L1_S5	0	0.5957487756928440	-0.030639516393344200	0.4637904192404770	0.7510158722362820
L1_S6	0	0.6191031249229520	-0.016667896079453300	0.48677051101027200	0.7468060265153140

Segmentation – Merging algorithm

- Region-growing Implementation
 - Input Data: List of the Segments/ Output Data: List of Region
 - 인근 segment들의 차이가 Threshold 이하면 하나의 region으로 합침



Segmentation – Merging algorithm

✓ Threshold 설정 기준

- 데이터 불균형 비율, merging 과정에서 발생한 error segment 비율, F1_score을 기준으로 비교
- 실제 사진 영역(5,000m 정사각형 버퍼 = 50ha) 대비 평균 피해면적(0.9ha) 비율 = $1/50 = 0.02$ 참고

Initial segmentation 결과

Non-Landslide segments 150532 개

비율 1:585

Merging algorithm 결과

Non-Landslide segments 3388 개

비율 1:13

Thr Coarse	Thr Fine	Landslide	Non-Landslide	Total Segments (Ratio)	Reduction	Error	F1_score
0.2	0.05	257	5527	5784 (1:21)	96%	64%	0.67
0.25	0.05	257	3388	3645 (1:13)	97%	72%	0.79
0.25	0.1	257	3300	3557 (1:12)	97%	72%	0.78

모델링 & 튜닝 - (1) X, y 변수

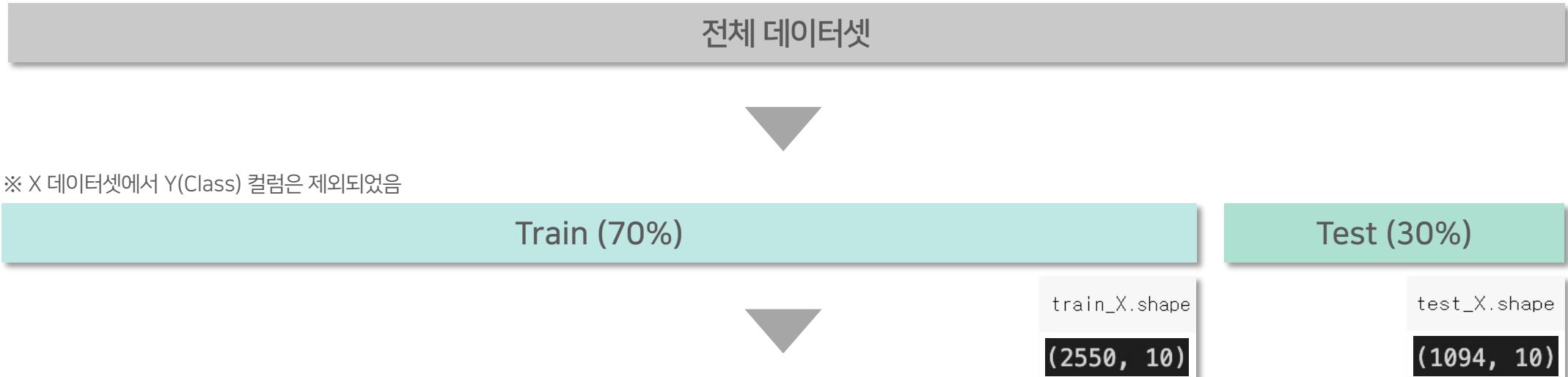
산사태 발생 여부를 예측하는 이진 분류 모델 학습을 위한 X, y 변수 설정

- X : 10개 변수
- Y : 산사태 발생 여부 (1: 발생, 0: 발생하지 않음)

X					y
NDVI	Slope mean	Slope max	ND std	Relative relief	
ratio rg change deviation	Brightness change deviation	NDVI change deviation	gNDVI deviation	Brightness	산사태 발생 여부

모델링 & 튜닝 – (2) train/test 데이터 분리

하나의 데이터셋을 train/test 데이터셋으로 분리



모델링 & 튜닝 – (3) 연속형 데이터

연속형 데이터 Standard Scaling

Train 데이터셋 scaling한 만큼 Test 데이터셋 scaling 진행

	ndvi	ratio_rg_change_deviation	brightness_change_deviation	ndvi_change_deviation	brightness	slope_mean	gndvi_deviation	slope_max	nd_std	relative_relief
328	0.143441	-0.083389	-0.008348	0.106329	0.776667	4.804399	-0.456119	21.811171	0.232277	7.000000
1789	0.088134	-0.102105	0.057645	0.216367	0.756953	12.289646	-0.500204	36.491138	0.192811	33.183381
2498	0.256566	-0.050054	0.027504	0.000326	1.067047	7.283115	-0.184062	15.970316	0.143464	16.000000

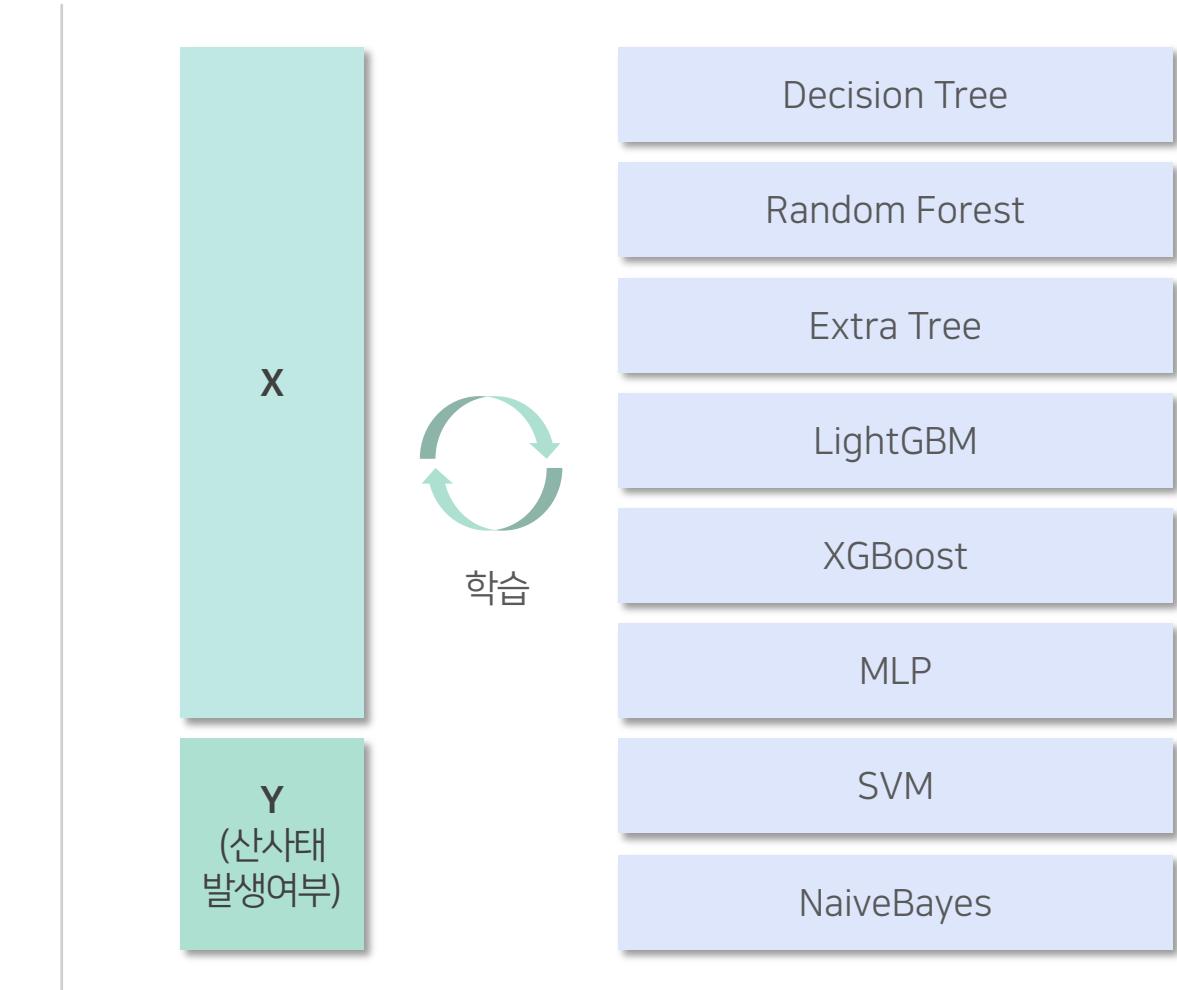


	ndvi	ratio_rg_change_deviation	brightness_change_deviation	ndvi_change_deviation	brightness	slope_mean	gndvi_deviation	slope_max	nd_std	relative_relief
328	-0.905999	-1.027011	0.590440	0.452266	-1.032413	-0.745566	-1.801453	-0.238603	2.454213	-0.723863
1789	-1.331062	-1.170534	0.958985	1.386956	-1.099521	0.025751	-2.086781	0.969437	1.546823	-0.074591
2498	-0.036578	-0.771375	0.790660	-0.448165	-0.043931	-0.490147	-0.040624	-0.719257	0.412217	-0.500689

모델링 & 튜닝 - (4) 모델 학습

산사태 발생 여부 예측을 위한 분류 모델 선택 및 학습 진행

모델	설명
Decision Tree	데이터 분류와 예측을 위한 트리 구조의 알고리즘으로, 데이터의 특성에 따라 의사결정 규칙을 생성하여 문제를 해결함
Random Forest	여러 개의 의사결정나무(Decision Tree)를 조합하여 안정적이고 정확한 예측을 수행하는 앙상블(Ensemble) 기법
Extra Tree	랜덤포레스트와 유사하지만 더 빠르게 학습하고, 과적합 방지에 더 적합하여 보다 좋은 성능을 보이는 모델
LightGBM	일반적인 그래디언트 부스팅(Gradient Boosting) 방법을 사용하면서도 빠른 학습과 예측 속도, 과적합 방지 등의 성능 향상을 위한 최적화 기법을 적용한 부스팅 알고리즘
XGBoost	트리 분류기와 부스팅 알고리즘을 활용하여 높은 예측 성능을 가지며, 과적합 방지와 빠른 학습 및 예측 속도 등이 특징인 알고리즘
MLP	인공 신경망 기반의 다층 퍼셉트론으로, 복잡한 비선형 문제를 다룰 수 있는 대표적인 머신러닝 알고리즘
SVM	데이터를 고차원 공간으로 매핑하여 클래스 간의 최대 마진을 찾아내는 방식으로 작동하는 대표적인 분류 머신러닝 알고리즘
Naive Bayes	베이즈 이론을 기반으로 하는 확률적 분류 알고리즘



모델링 & 튜닝 - (5) 모델 하이퍼파라미터 튜닝

Grid Search CV를 진행하여 최적의 하이퍼파라미터 선정

Random Forest	LightGBM	XGBoost
<pre># Define the parameter grid to search param_grid = { 'n_estimators': [50, 100], 'max_depth': [5, 10], 'min_samples_split': [5, 10], 'min_samples_leaf': [1, 2, 4] }</pre>	<pre># Define the hyperparameters to tune param_grid = { 'num_leaves': [5, 10], 'learning_rate': [0.001, 0.01, 0.1], 'n_estimators': [100, 500, 1000] }</pre>	<pre># Define the hyperparameters to be tuned param_grid = { 'n_estimators': [50, 100, 200], 'max_depth': [3, 5, 7], 'learning_rate': [0.001, 0.01, 0.1] }</pre>
Decision Tree	MLP	Extra Tree
<pre># Define the hyperparameters to tune param_grid = { 'depth': [3, 5, 7, 10, 20], 'learning_rate': [0.001, 0.01, 0.1, 0.3, 0.5] }</pre>	<pre># Define the hyperparameters to be tuned param_grid = { 'hidden_layer_sizes': [(10,), (50,)], 'activation': ['logistic', 'tanh', 'relu'], 'solver': ['lbfgs', 'sgd', 'adam'], 'alpha': [0.001, 0.01, 0.1] }</pre>	<pre># Define the parameter grid to search param_grid = { 'n_estimators': [50, 100], 'max_depth': [5, 10], 'min_samples_split': [5, 10], 'min_samples_leaf': [1, 2, 4] }</pre>

최적 모델 선정

모델 비교

F1-score: 불균형 데이터에서 잘 작동하는 평가지표

평가 지표 \ 모델	Decision Tree	Random Forest	Extra Tree	LGBM	XGBoost	MLP	SVM	Naïve Bayes
F1 score (Default)	0.73	0.81	0.84	0.88	0.86	0.80	0.63	0.67
F1 score (Tuning)	0.74	0.82	0.76	0.85	0.85	0.80	-	-

최적 모델 선정

LightGBM의 F1-score가 가장 높으므로,

산사태 발생 예측 모델로 LightGBM 선정

최적 모델 선정

모델 비교

평가 지표	모델
F1 score (Default)	LGBM 0.88
F1 score (Tuning)	0.85

✓



classification_report

F1-score: 불균형 데이터에서 잘 작동하는 평가지표

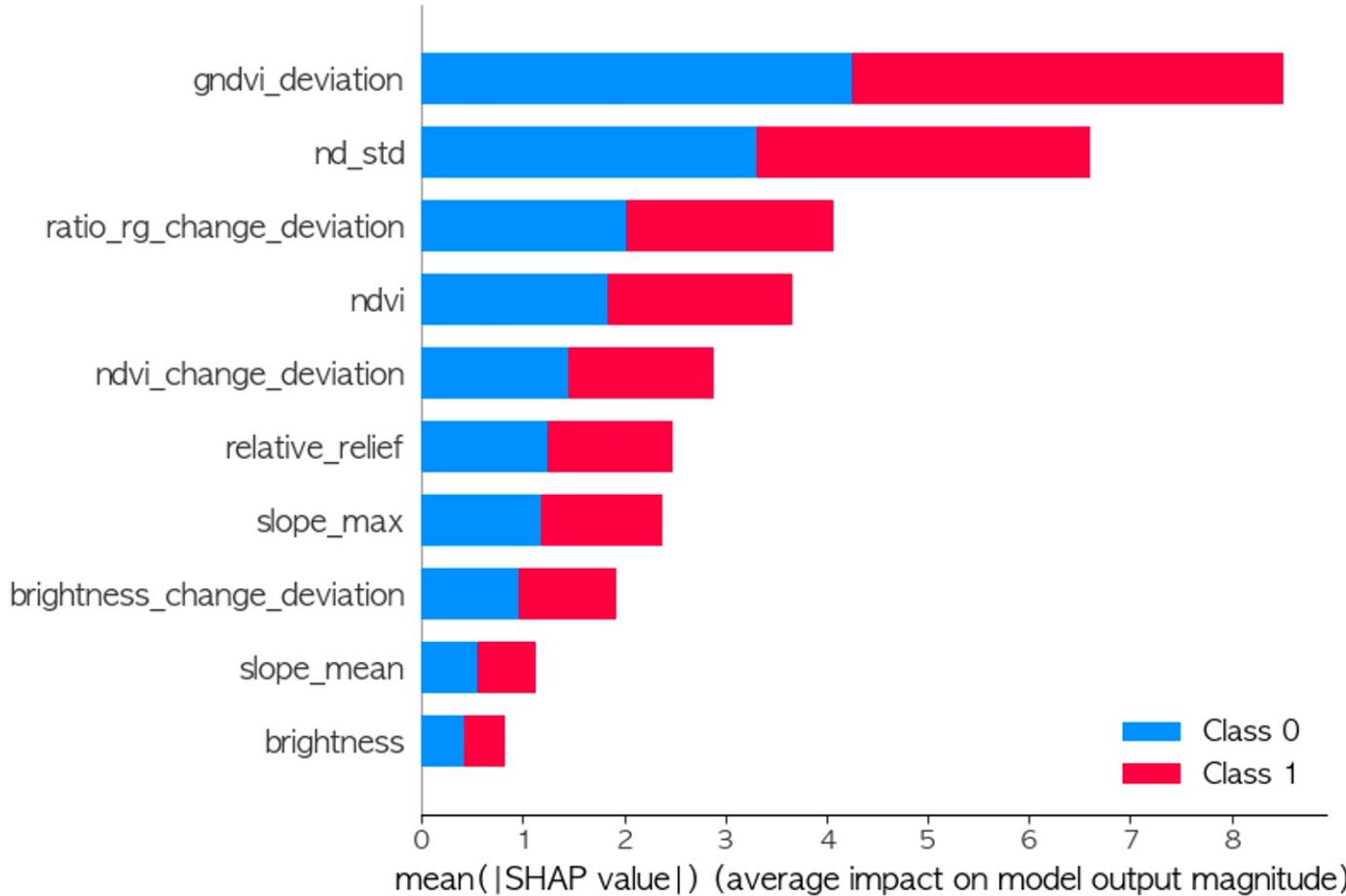
	precision	recall	f1-score	support
0	0.98	1.0	0.99	1009
1	0.96	0.81	0.88	85
accuracy			0.98	1094

최적 모델 선정

LightGBM의 F1-score가 가장 높으므로,
산사태 발생 예측 모델로 LightGBM 선정

변수 중요도 해석 - SHAP Feature Importance

SHAP 변수 중요도



Shap (SHapley Additive exPlanations)

: ML 모델에서 각 변수를 하나씩 빼고 더하면서 해당 변수가 y 예측의 정확도에 얼마나 기여하는지 계산하여 각 변수의 중요도를 수치화하는 방법

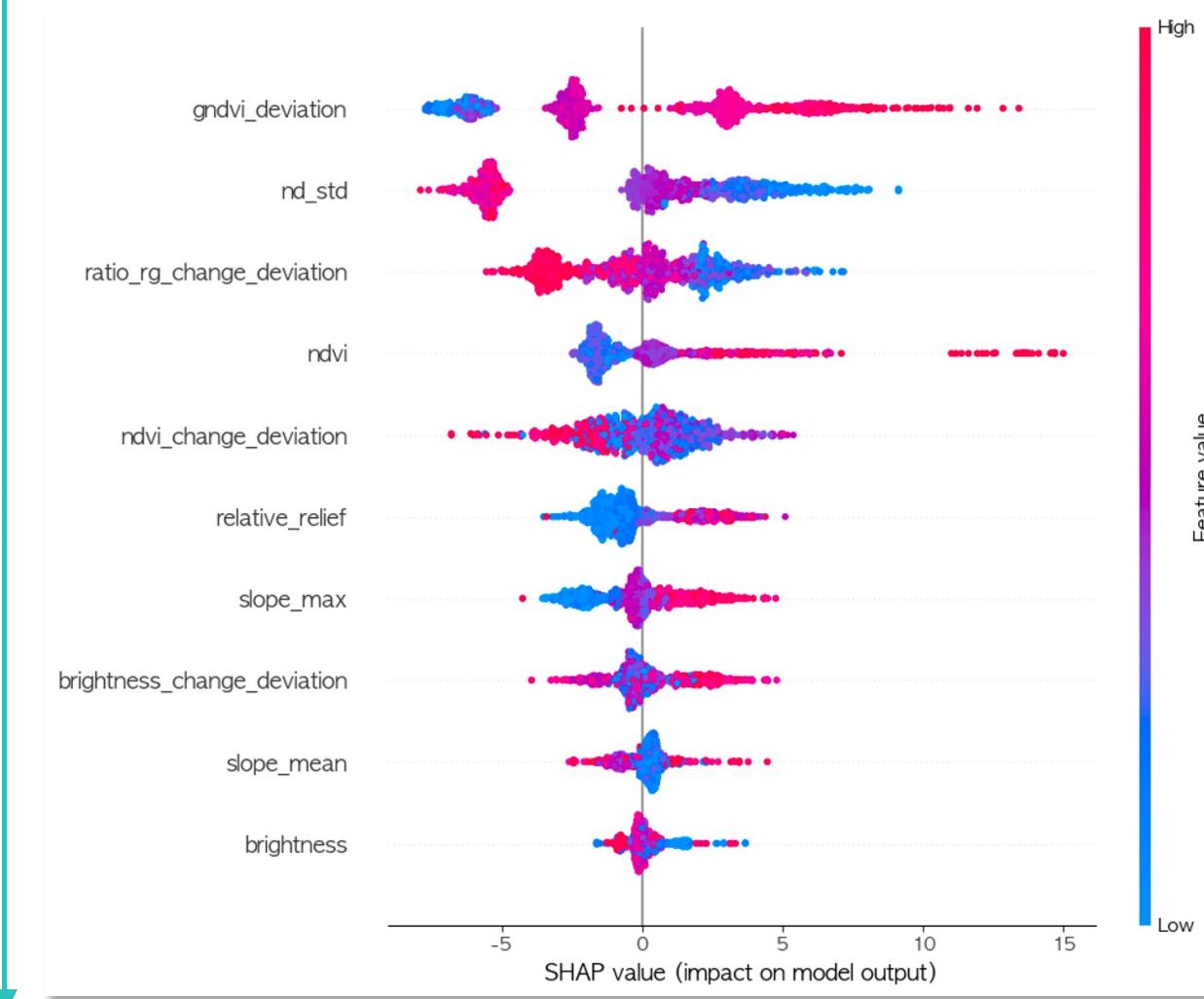
$$I_j = \sum_{i=1}^n |\phi_j^{(i)}|$$

산사태 감지에 중요한 변수 파악

산사태 전조 증상 파악

변수 중요도 해석 - SHAP Summary Plot

변수 중요도 기준내림차순



변수별 SHAP value 해석

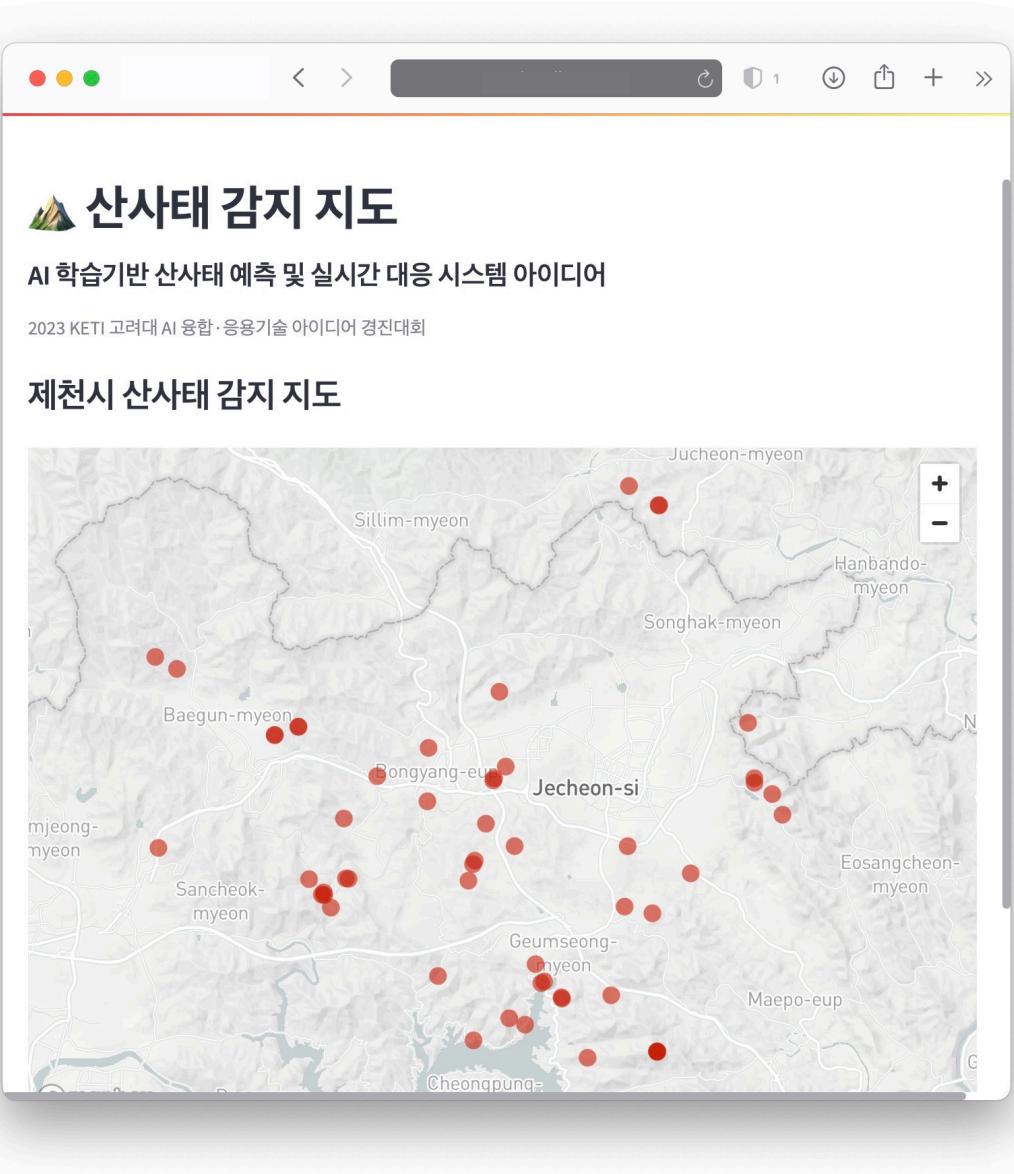
예측 기여 경향성	변수
값이 커지면 산사태가 발생하는 경향이 있는 변수들	<ul style="list-style-type: none"> • gNDVI Deviation • NDVI • Relative relief • Slope max
값이 커지면 산사태가 발생하지 않는 경향이 있는 변수들	<ul style="list-style-type: none"> • NDVI std • Ratio RG change deviation • NDVI change Deviation

Part 4

실시간 대응 시스템 아이디어



분석 결과 시각화

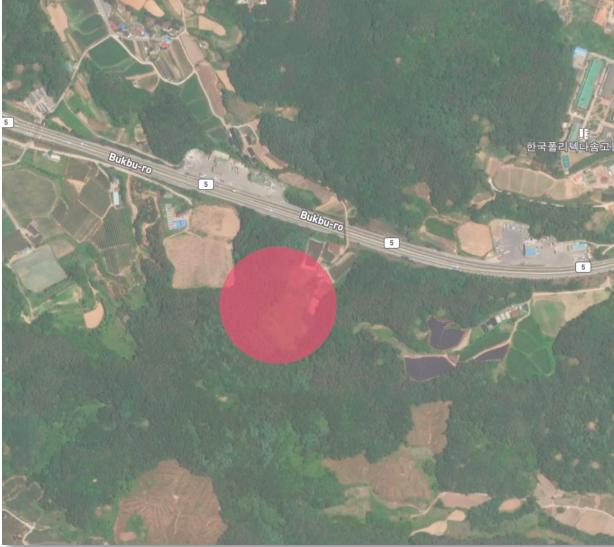


- Streamlit으로 산사태 예측 결과 시각화
- 추후 산림청 및 교통정보센터에서 참고



분석 결과 시각화

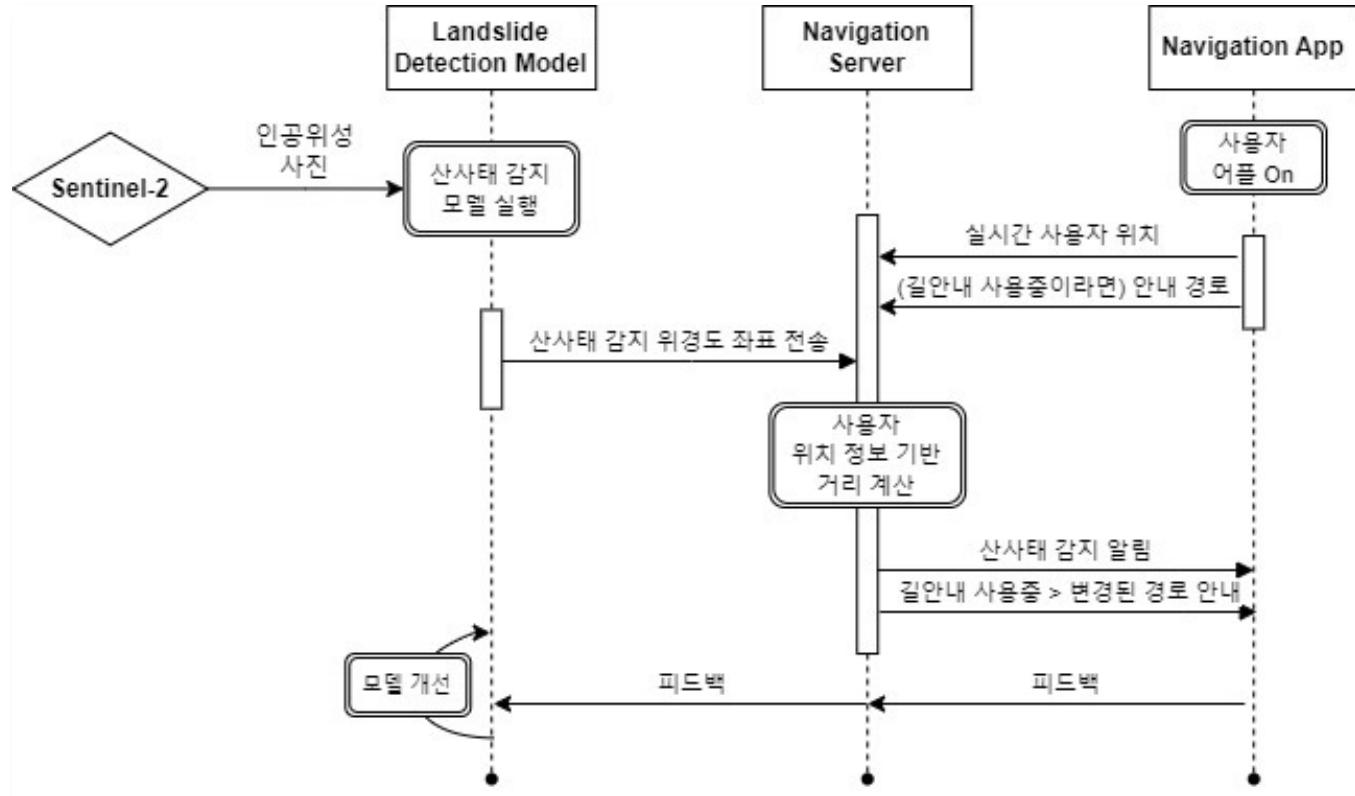
도로변 산사태 감지 지역



산 주변, 도로변에서 많은 산사태가 감지됨

추후 운전자에게 내비게이션, 알람을 통해 전달

Message Sequence Chart (MSC)

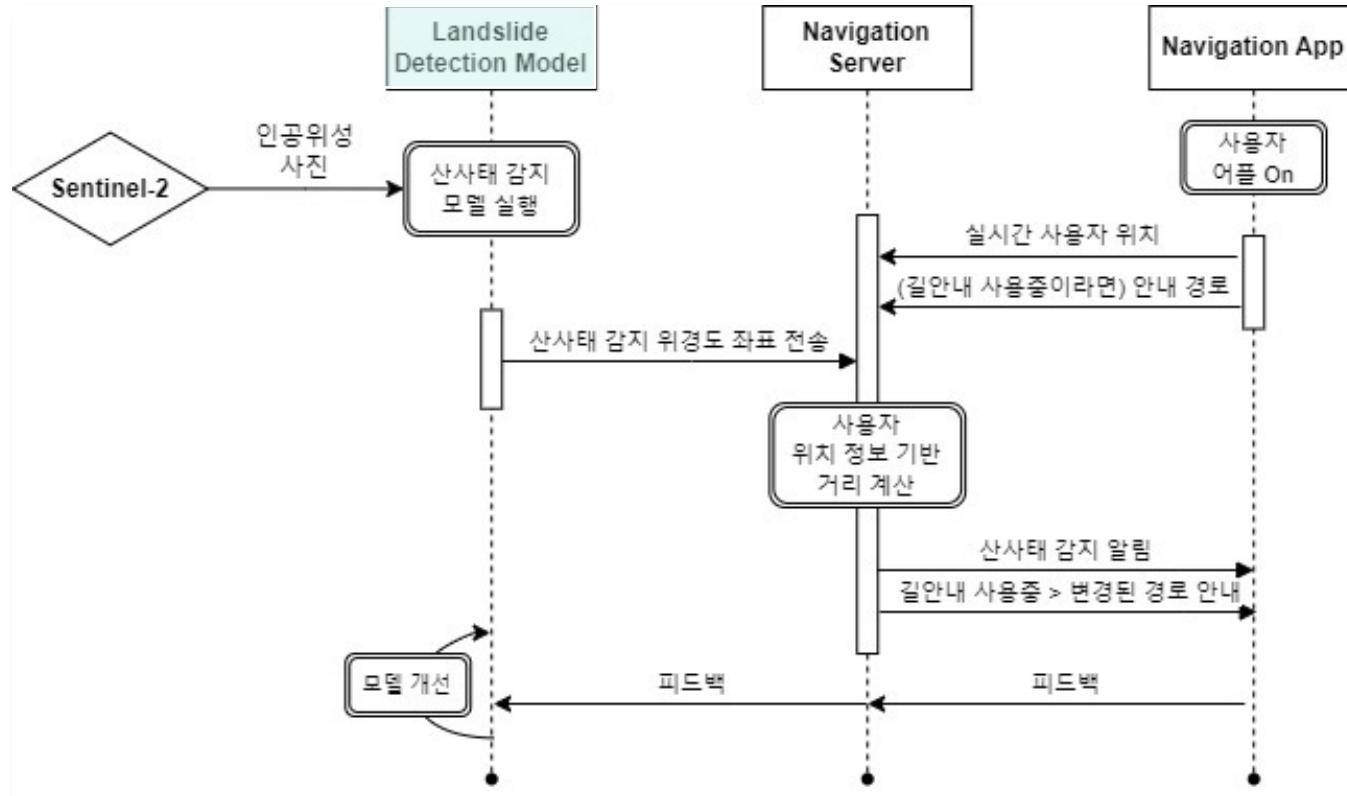


Landslide Model

Navigation Server

Navigation App

Message Sequence Chart (MSC)



Landslide Model

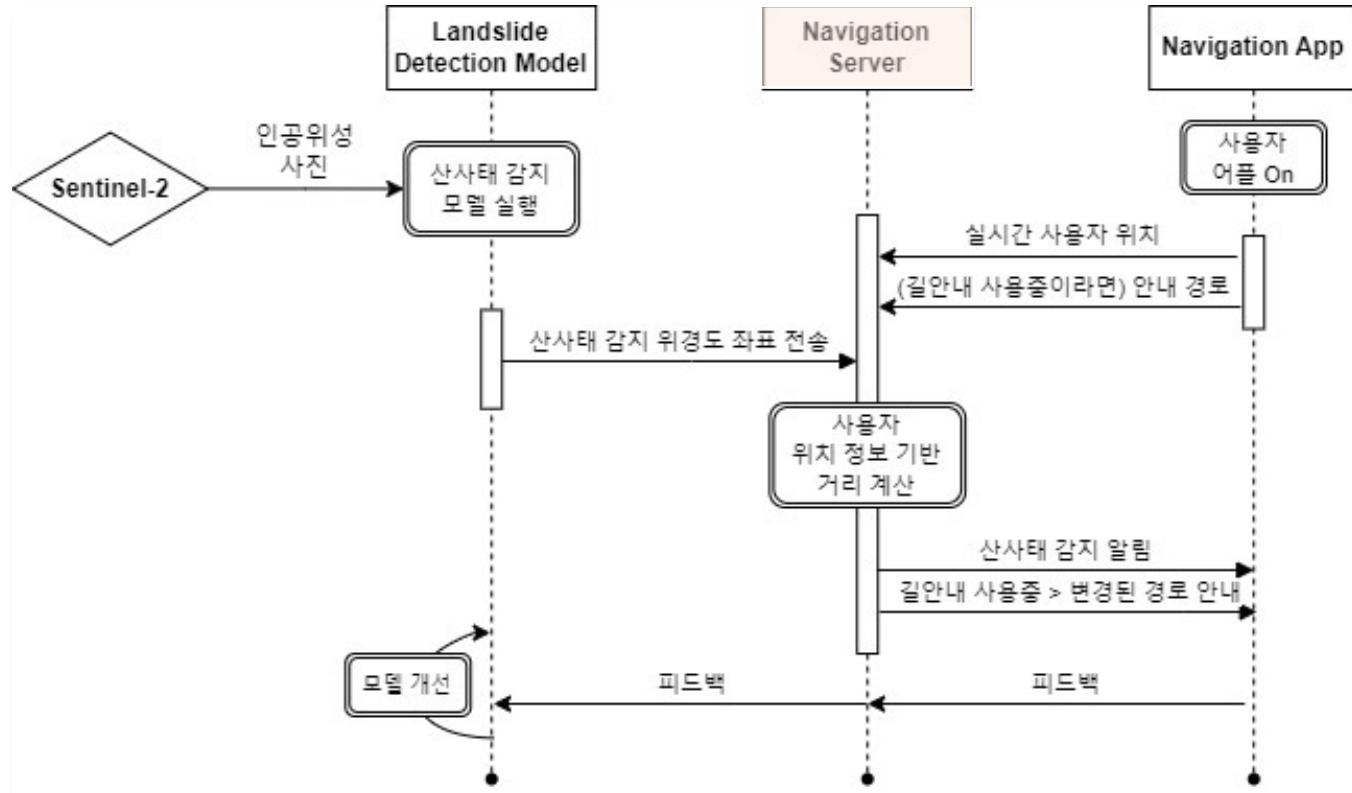
Sentinel-2 를 통해 위성 사진 수집

위성 사진을 input하여 Detection model 실행

산사태 감지된 위경도 좌표 Server로 전송

서비스에 대한 사용자 피드백 통해 모델 개선

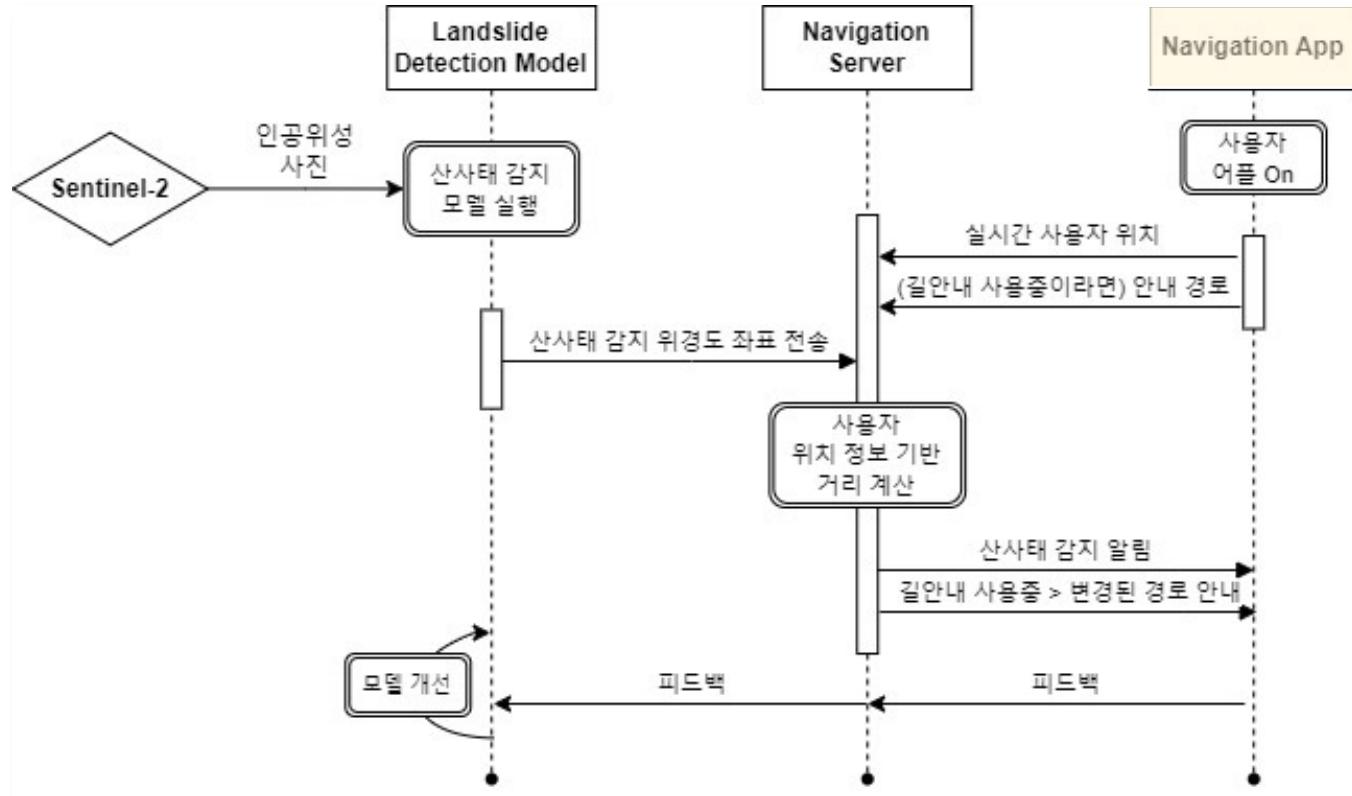
Message Sequence Chart (MSC)



Navigation Server

- 산사태 감지 좌표,
실시간 사용자 위치 및 길안내 경로 정보 수합
- ↓
- 사용자 위치 정보 기반 거리 계산
- ↓
- App으로 산사태 감지 전송
- ↓
- 사용자가 길 안내 이용중이면,
App으로 변경된 경로 전송

Message Sequence Chart (MSC)



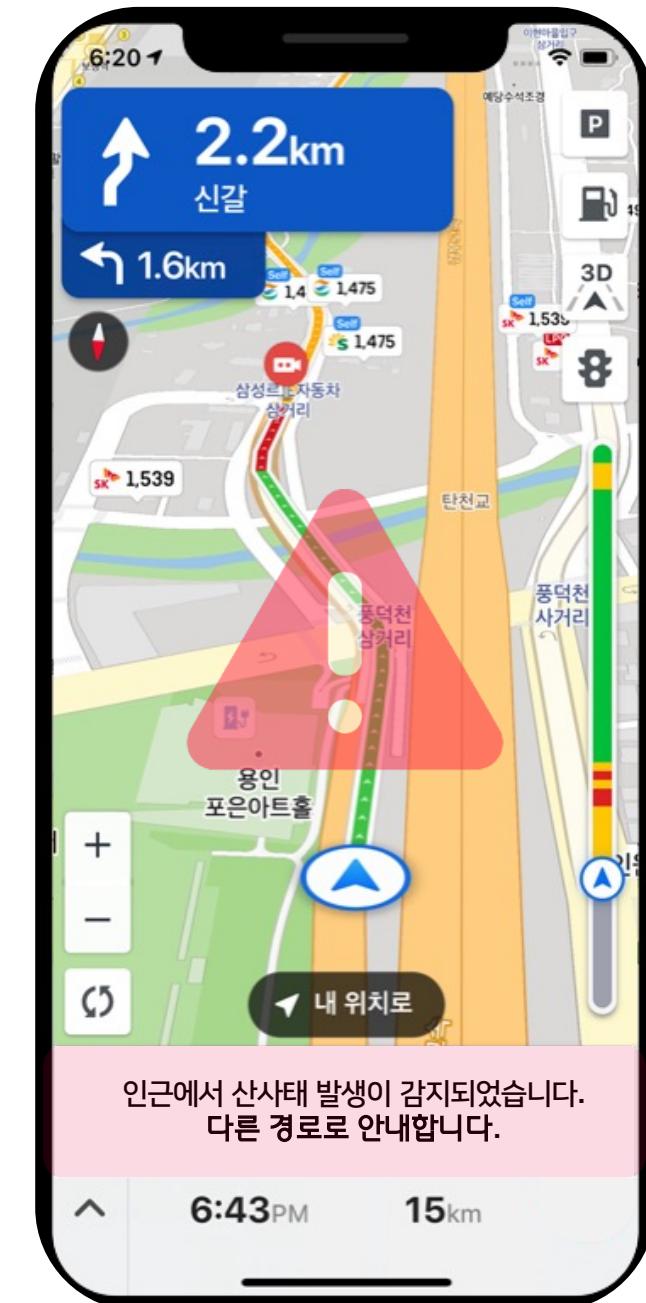
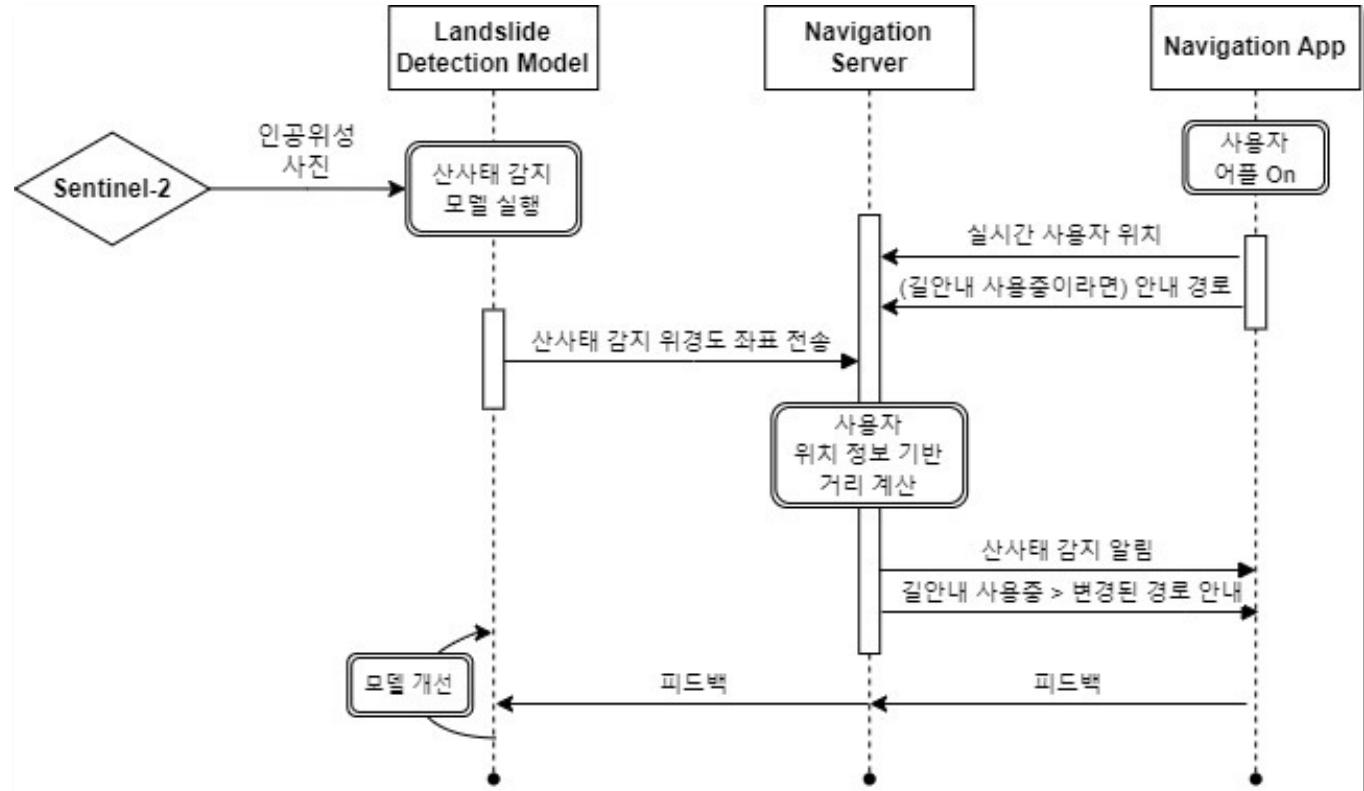
Navigation App

사용자에게 산사태 감지 알림

사용자가 길 안내 이용중이면, 변경된 경로 안내

사용자로부터 서비스 피드백 받아 서버로 전송

MSC 활용방안



Part 5

결론



기대효과



실시간 산사태 감지 모델 개발

1



실시간 산사태 감지 정보를
산림 및 도로 관계자에게
전달하여 신속한 대응 가능

2



내비게이션 통한
실시간 산사태 경보서비스 제공
및 정보 반영한 추천 경로 안내

3



자율주행 자동차의
안전한 경로 선택 가능 &
자연재해 대응으로 인해
자율주행 기술의 신뢰성 향상

의의 및 발전 방향



의의

단순 지형, 날씨 등 산림입지환경 변수에 따른 산사태 예측이 아닌 실시간 감지 및 이후 대응에 맞춰진 모델링

산사태 이후 발생 가능한 사건으로 인한 피해 최소화

변수 중요도 해석을 통해 산사태 감지에 중요한 변수를 파악



발전 방향

지형, 날씨 등의 변수들을 추가 및 변수 선택하여 모델의 성능 향상

정확한 산사태 발생일자를 가진 데이터 확보한다면 더 정교한 모델 구현 가능

Part 6

툴·데이터 및 참고문헌



활용 툴 및 참고문헌 출처



활용 툴	활용 분야
JAVA	모델링에 필요한 이미지 획득
Google Earth Engine	모델링에 필요한 이미지 획득
Python (pandas, numpy, scikit-learn, matplotlib, seaborn, shap등)	데이터 시각화, 분석, 모델링

논문

Meylin Herrera Herrera. (2019). Landslide Detection using Random Forest Classifier

Github

https://github.com/mhscience/landslides_detection

2023 KETI 고려대 AI 융합·응용기술 아이디어 경진대회
AI 학습기반 산사태 예측 및 실시간 대응 시스템 아이디어

감사합니다.

고려대학교 통계학과

원윤정 (jennie0715@korea.ac.kr)
김혜연 (yeorii@korea.ac.kr)
장희중 (vicky4150@korea.ac.kr)



Part 7
부록



이미지 추출과 전처리

Google Earth Engine을 활용한 이미지 데이터 추출과 전처리

Algorithm 3.1: Landslide Image Pre-processing (GEE)

```

Input: satellite, landslides
1 Initialize image collector
2 for landslides do
3   Add landslide to new before Processor
4   Add landslide to after Processor
5 Process Before (landslide):
6   calculate event time range before event
7   check for cloud free images in time range
8   apply cloud masking
9   add to image collection
10 Process After (landslide):
11  calculate event time range
12  check for cloud-free images in time range after event
13  apply cloud masking
14  add to image collection
15 Process Collector:
16  if image collection is ready then
17    calculate band rationing
18    normalize the image bands Red,Green,Blue
19    calculate NDVI
20    calculate Brightness
21    calculate image difference for NDVI , Brightness, band rationing
22    download image difference as .tiff
23 end

```

이미지 전처리

Homogeneity criterion에 기반한 Decision

Algorithm 3.2: Decision based on homogeneity criterion.

Input: region, neighbor, NDVI non-local mean , NDVI non-local std, VID
non-local, VID non-local std, RGD non-local

Output: True/False

```

1 set coarse-grained threshold
2 set fine-grained threshold
3 calculate similarity criteria ( $Sm$ ) using NDVI
4 right VID boundary = vid mean + 2 std
5 right RGD boundary = rgd mean + 2 std
6 if  $NDVI_{neighbor} > NDVI_{non-local\ mean}$  AND  $Sm < coarse-grained\ threshold$ 
    then
    7   return True
8 if  $VID_{neighbor} < right\ boundary\ VID$  AND  $Sm < coarse-grained\ threshold$ 
    then
    9   return True
10 if  $RGD_{neighbour} < right\ RGD\ boundary$  AND  $Sm < coarse-grained\ threshold$ 
    then
    11   return True
12 if  $Sm < fine-grained\ thershold$  then
    13   return True
14 return False

```

이미지 전처리

이미지 세그먼트 수준에서 Merging Algorithm을 통해 Region으로 merged

Algorithm 3.3: Region growing at segment level

Input: list of segments (Kd-tree data structure)

Output: list of regions

```
1 while list of segments is not empty do
2   seed = []
3   region = []
4   find the segment with minimum RGD
5   add the segment to the seeds list
6   add the segment to the regions list
7   remove segment from the segments list
8   while seeds do
9     find the neighbors of every seed segment;
10    for every neighbor do
11      if similarity measure gives true then
12        add current neighbor to the region;
13        remove current point from the segment list;
14        add neighbor to the seeds list
15 return list of regions
```
