

# 생태계를 고려한 절차적 지형 생성

## Procedural Terrain Generation considering the ecosystem

2022/07/24

조선대학교 컴퓨터공학과, 20171614 노순건

### I. 서 론

게임은 콘텐츠를 소모하는 응용 프로그램으로, 콘텐츠의 고갈은 곧 플레이어의 감소를 의미한다. 이는 개발자의 수익감소로 이어지므로 개발자들은 콘텐츠 고갈을 막기 위해 콘텐츠 추가 업데이트를 한다.

이렇게 업데이트하는 방식 대신 새로운 콘텐츠를 동적으로 제공해주는 절차적 생성 알고리즘은 2000년대 중반부터 지금까지 게임 산업에서 각광받는 기술로 일련의 규칙을 반복적으로 수행하여 콘텐츠를 생성해준다[1]. 이를 통해 개발자에게는 게임 콘텐츠 생성에 필요한 수고를 덜어주고 플레이어들한테는 다양한 경험을 제공해준다. 이 기술을 사용한 대표적인 게임으로는 마인크래프트와 노 맨즈 스카이가 있으며, 마인크래프트는 절차적으로 무한한 맵을 노 맨즈 스카이는 경 단위의 행성을 절차적으로 생성한다.

이처럼 거대한 지형생성은 일반적으로 자연스럽게 정렬된 난수인 펄린 노이즈를 사용한다[2]. 노이즈에서 생성되는 그레이스케일 이미지를 높이 정보로 삼아 평평한 면에서 3D 메쉬로 렌더링하여 현실 세계의 지형을 모방하는 것이다.

최근에는 이렇게 무엇인가를 모방하는 문제를 기계학습으로 해결해보고자 하는 연구가 활발하게 수행되고 있다[15]. 특히 GANs는 어떠한 이미지 데이터 세트 사이에서 특징들의 분포를 학습하여 그 분포를 모방한 진짜 같은 이미지를 생성한다.

이러한 연구들이 진행되면서, 지형 위성사진 데이터를 이용하여 GANs 모델을 학습시켜, 현실 세계지형을 모방한 지형을 생성해보고자 하는 연구 또한 수행되어졌다[2, 3, 4].

이에 이 프로젝트에서는 먼저 수행된 연구와는 다른 데이터 전처리 방식을 제안하여 지형을 생성해보고자 한다. 또한, 다양한 지형 환경의 데이터를 수집하는 방식을 제안하여 다양한 생태계를 고려한 지형 데이터를 라벨과 함께 GANs 모델을 학습시켜 지형을 생성해보고, 학습된 모델의 결과물을 시각화하는 응용 프로그램을 구현해보고자 한다.

2장에서는 관련된 연구들에 대해서 살펴본다. 3장에서는 문제를 해결하는 접근법에 대해서 살펴본 후, 개발 일정과 구현 환경 및 전체 구성도, 데이터 수집 및 전처리에 관한 내용과 학습시킬 모델에 대해 살펴본다. 4장에서는 학습 결과와 이를 시각화하는 응용 프로그램 구현 및 3D 시각화 결과를 확인한다. 마지막으로 5장에는 결론 및 고찰과 느낀점에 대해 서술하였다.

## II. 관련 연구

### 2.1 GANs을 사용한 절차적 3D 지형 생성

GANs을 사용하여 3D 지형의 구성요소인 색을 나타내는 Texture와 높이를 나타내는 Height map을 생성하는 두 연구를 찾을 수 있었다[3, 4].

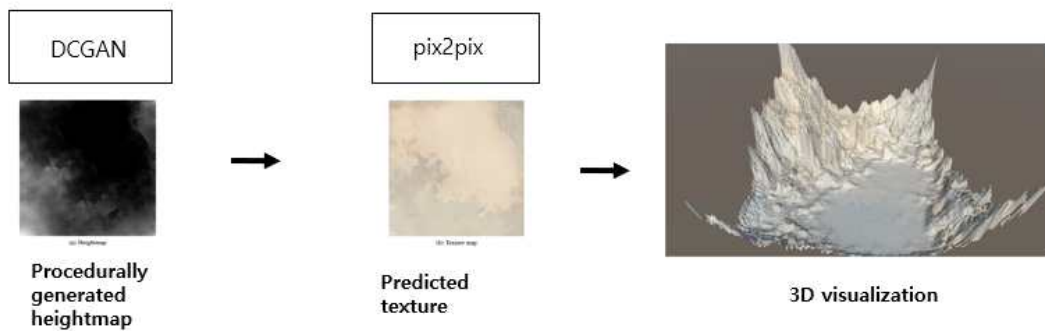


그림 1. A step towards procedural terrain generation with GANs [3]

NASA가 제공하는 위성사진과 높이를 나타내는 DEM(Digital Elevation Model) 데이터를 사용한 연구[3]는 그림 1에서 보여지듯, DEM 데이터들을 DCGAN 모델로 학습시켜 DEM 데이터를 생성한 후, DEM과 위성사진으로 학습시킨 pix2pix 모델로 Texture를 예측하여 3D 지형을 생성하였다. 여기서 DEM 데이터는 높이를 나타내는 1채널 그레이스케일 이미지 데이터로 3D 지형 요소인 Height map에 대응된다.

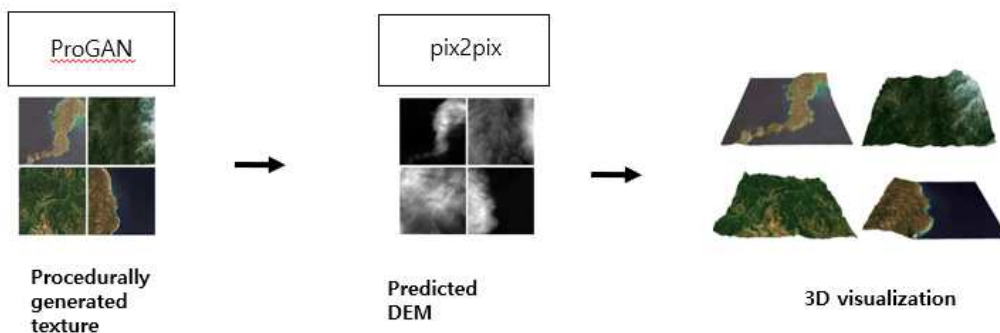


그림 2. Procedural 3D Terrain Generation using GANs [4]

ALOS World 3D 30m의 데이터 DEM 과 Google Earth Engine의 위성사진을 이용한 연구 [4]는 그림 2에서 보여지듯, 위성사진 데이터들을 ProGAN 모델로 학습시켜 위성사진을 생성한 후, DEM과 위성사진으로 학습시킨 pix2pix로 DEM을 예측하여 3D 지형을 생성하였다.

두 연구의 결론 및 향후 계획은 다음과 같다. 연구[3]에서는 두 개의 모델을 같이 훈련시키지 않았기에 일부 텍스처의 일부가 완전히 흰색으로 나타나는 문제가 생겼다고 저자들은 추측했다. 연구[4]에서는 두 개의 모델의 범위를 결합한 글로벌 모델을 구현 즉, 그럴듯한 DEM을 생성하면서 임의의 위성 이미지를 생성해야 할 필요성을 말하였다. 이런 글로벌 모델은 두 문제에 대한 결합된 손실을 최소화해야 하므로 수렴하는 것은 어려울 수 있으나 더 현실적이고 강력한 결과를 얻을 수 있다고 하였다[4]. 이처럼 두 연구[3, 4] 모두 모델들을 결합하여 두 문제를 해결할 하나의 앙상블 모델을 구현하면 성능이 향상될 것이라고 결론지었다.

## 2.2 RGB 데이터와 Depth 데이터를 사용한 연구

2.1에서 볼 수 있듯이 3D 지형생성은 RGB 데이터와 Elevation 즉, Depth 데이터를 학습시켜 문제를 해결해야 한다. 이와 같은 데이터를 다루는 관련 연구들은 다음과 같다.

헤드 마운트 디스플레이 제거를 위한 연구[6]는 VR기기 등으로 가려진 얼굴 이미지에서 누락된 색과 깊이 정보를 복원시키는 연구이다. 이 연구[6]에서는 이를 연구하기 위해 얼굴 이미지 RGB와 깊이 정보 Depth를 하나의 RGB-D 즉, RGBA 파일로 전처리하여 문제를 해결하는 방안을 제시하였다.

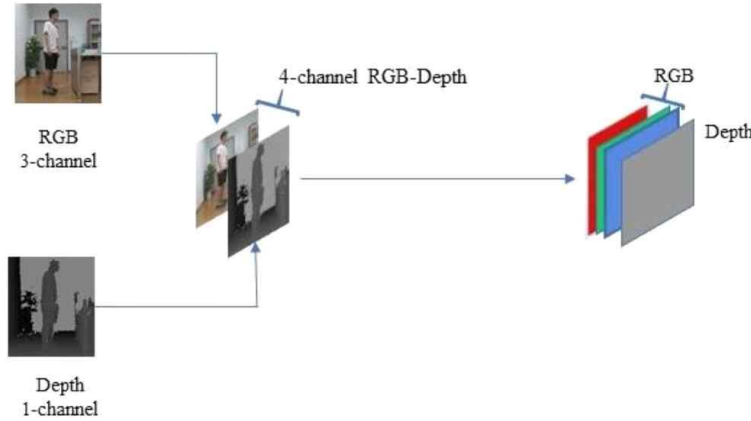


그림 3. Fusion in dissimilarity space for RGB-D person re-identification [5]

RGB-D 인물 재식별을 위한 이질성 공간 융합 연구[5]는 사람 재식별을 위해 그림 3에서 보여지듯이 RGB와 Depth 데이터를 연결하여 RGBA 데이터 셋을 구축하였고, 다양한 RGB-D 기반 응용 프로그램에서의 문제를 해결하는 방안을 제시하였다.

$$x' = x \oplus M_x = \{x_r, x_g, x_b, g(M_x; \theta)\} \quad (1)$$

이미지 대 이미지 변환을 위한 주의 기반 공간 안내 연구[7]는 주의 맵을 제공하여 실제 이미지와 가짜 이미지를 구별하기 위한 정보 영역을 같이 학습시키는 방안을 제시하였다. 이 연구[7]에서는 샘플 이미지를 수식 (1)의  $x_r, x_g, x_b$ 에 할당하고, 각 픽셀의 중요도를 명시적으로 나타내는 주의 맵을 수식 (1)의 알파 채널에 넣어 하나의 RGBA 데이터로 전처리하여 문제를 해결하였다.

즉, 관련 연구들[5, 6, 7]은 주어진 문제를 해결하기 위해 이미지와 추가적인 하나의 데이터를 결합하여 하나의 RGBA 데이터로 전처리한 것이다.

## 2.3 GANs

GANs(Generative Adversarial Networks)[8]는 적대적 프로세스를 통해 생성 모델을 추정하기 위한 프레임워크이다. 이는 데이터 분포를 추정하는 생성자 G와 샘플을 구별하는 판별자 D로 구성된다. 생성자는 실제 이미지 데이터 셋의 분포를 학습하며 판별자는 동시에 실제 이미지 데이터 셋에서 나왔을 확률을 학습한다.

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(x)} [\log (1 - D(G(z)))] \quad (2)$$

이들의 목적함수는 수식 (2)에서 볼 수 있듯이 minmax 알고리즘에 기초하여 G에 대한 학습은 D가 실수할 확률을 최대화하는 것 즉, 목적함수를 최소화하기 위해  $D(G(z))$ 가 1이 되도록 하는 것이 목적이며, D에 대한 학습은 두 log의 결과가 0으로 하는 것 즉, 목적함수를 최대화하는 것이 목적이다.

## 2.3 ProGANs

2.1의 연구[4]에서 Texture를 생성하는 모델인 ProGANs(Progressive Growing of GANs) [9]는 2.4의 GANs의 생성자와 판별자를 그림 4와 같이 저해상도에서 고해상도 이미지로 점진적으로 학습시키는 것이다. 이로 인해 빠르고 안정적으로 학습할 수 있으며 고해상도의 이미지를 생성할 수 있다.

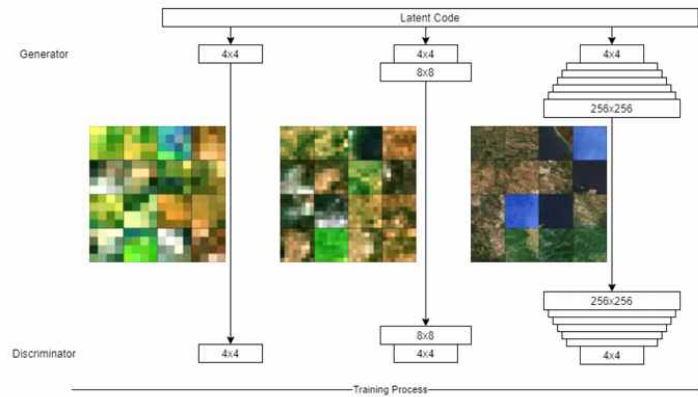


그림 4. 연구[4]의 ProGANs

## 2.4 CGANs

$$\min_G \max_D V(D, G) = \mathbb{E}_{\mathbf{x} \sim p_{\text{data}}(\mathbf{x})} [\log D(\mathbf{x}|\mathbf{y})] + \mathbb{E}_{\mathbf{z} \sim p_z(\mathbf{z})} [\log(1 - D(G(\mathbf{z}|\mathbf{y})))] \quad (3)$$

CGANs(Conditional Generative Adversarial Networks)[10]은 GANs[8]의 확장으로 추가적인 정보인 클래스 라벨 또는 추가 정보를 같이 학습시킬 수 있는 신경망이다. 이는 목적함수 (2)에 조건 변수  $y$ 를 넣은 목적함수 (3)에 의해 수행된다.

## 2.5 종분포모형(SDM)과 서식지 분포

생태계란 생물 종이 살아가는 기반이 되는 환경을 의미한다[11]. 이러한 생태계에서 일어나는 변화나 평가를 위해 공간 데이터와 환경 데이터를 통계적 기법이나 모형화를 통해 외삽하는 방법들을 사용하는데, 종분포모형이나 서식지분포모형이 대표적이다[12].

종분포모형은 앙상블 모형[13]이나 GIS와 로지스틱 회귀[13], Maxent[14] 등의 다양한 기계학습 기반으로 수행되며, 학습에 필요한 데이터는 종 출현/비출현 데이터와 기후환경 데이터 그리고 토지환경 데이터 등이 있고[14], 학습된 모델에 변수를 넣어 서식지 분포 확률을 얻어내어 생물 다양성 평가나 보호지역으로 지정하는 등 서식지 관리 및 복원을 위한 자료로 활용한다.

## III. 생태계를 고려한 절차적 지형생성

2.1에서 살펴본 지형생성 연구들[3, 4]은 하나의 앙상블 모델 대신 데이터 전처리를 통해 하나의 모델로 성능을 높일 수 있을 것이라고 하였다. 하지만 이 프로젝트에서는 2.2에서 살펴본 관련 연구들[5, 6, 7]처럼 학습에 사용되는 데이터인 위성사진과 DEM을 하나의 데이터로 전처리한후, 이 데이터들로 모델을 학습시키고자 한다.



그림 5. 두 데이터를 하나의 RGBA 데이터로 전처리

그림 5에서 보여지듯이, 위성사진은 RGB 채널, DEM은 그레이스케일 이미지이므로 Alpha 채널에 해당한다.

그리고, 이 프로젝트에서는 2.1에서 살펴본 연구[4]에서 구현된 ProGANs[9]를 토대로 C GANs[10]을 적용하여 다양한 환경의 지형 데이터에 라벨을 설정한 후, 무작위적으로 생성되는 환경의 지형이 아닌 사용자가 원하는 환경의 지형이 생성되도록 구현하고자 한다.

또한, 다양한 환경의 지형 데이터를 수집하는 과정에서 2.5에서 살펴본 중분포모형(서식지 분포)을 사용하는 방안을 제안하고자 한다.

### 3.1 개발 일정

프로젝트 개발일정은 아래와 같이 진행되었다. 프로젝트 주제가 변경된 4, 5일은 제외하였다. 또한, 자료조사, 예제 코드 작성, 테스트 모델 학습 등과 같은 내용은 제외하고 핵심적인 내용만 표시하였다.

추진 내용	수행기간(일) (표시 : ■)														
	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
데이터 수집 및 전처리	■	■	■	■	■	■	■	■	■	■	■				
ProGANs 수정											■	■	■	■	
모델 학습														■	■
응용 프로그램 구현														■	■

### 3.2 개발 환경

개발 환경은 아래와 같다.

분류	이름
SW	운영체제 Windows 11 Education 64bits (22H2.1)
	개발언어 Python 3.10, R4.2.1, C++ 17
	프레임워크 Pytorch1.12 (CUDA 11.3)
	API Google Earth Engine, OpenGL
	DB Redis7.0
HW	GPU GeForce GTX 1660 super, 6GB
	CPU 인텔 i5-10400 CPU @ 2.90GHz
	MEM 32G, 2666Hz

### 3.3 전체 시스템 구성도

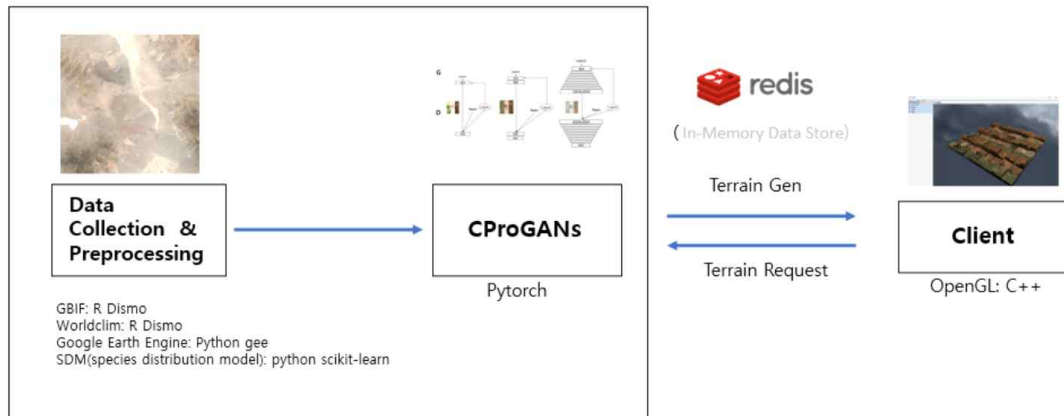
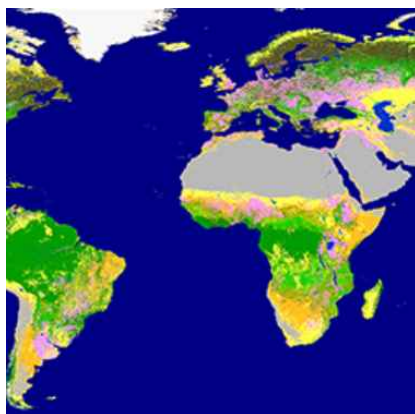


그림 6. 전체 시스템 구성도

전체 시스템 구성도는 그림 6에서 보여지듯이, 크게 3개의 시스템으로 나눌 수 있다. 파이썬과 R의 몇몇 패키지와 라이브러리로 데이터를 수집하고 전처리하는 시스템, 수집한 데이터를 통해 Pytorch 프레임워크로 모델을 학습시키는 시스템, 그리고 학습된 모델에 지형을 요청하고 지형을 받아 OpenGL API 기반으로 렌더링하는 시스템으로 구성된다.

### 3.4 데이터 수집 방법

여기서는 다양한 환경의 지형 데이터를 수집하는 방식 두 가지를 살펴보고, 그 중 서식지 확률 분포를 사용하여 데이터를 수집하는 과정을 설명한다. 먼저 첫 번째로 제안하는 방식은 그림 7에 해당하는 토지 피복도를 이용하는 방식이다.



Value	Color	Description
0	282828	Unknown. No or not enough satellite data available.
20	FFB22	Shrubs. Woody perennial plants with persistent and woody stems and without any defined main stem being shrub foliage can be either evergreen or deciduous.
30	FFFF4C	Herbaceous vegetation. Plants without persistent stem or shoots above ground and lacking definite firm str cover is less than 10 %.
40	F096FF	Cultivated and managed vegetation / agriculture. Lands covered with temporary crops followed by harvest (e.g., single and multiple cropping systems). Note that perennial woody crops will be classified as the appropriate land cover type.
50	FA0000	Urban / built up. Land covered by buildings and other man-made structures.
60	B4B4B4	Bare / sparse vegetation. Lands with exposed soil, sand, or rocks and never has more than 10 % vegetated in the year.
70	F0F0F0	Snow and ice. Lands under snow or ice cover throughout the year.

그림 7. Landcover data

토지 피복도는 Google Earth Engine에서 제공하는 데이터 셋 “COPERNICUS/Landcover/100m/Proba-V-C3/Global”를 사용하여 수집할 수 있다. 숲, 사막, 저수지 등과 같은 환경의 상태를 색으로 표현하는 토지 피복도는 산림 모니터링, 작물 모니터링, 생물양성 및 보전, 아프리카의 환경 등 생태계의 상태를 파악하는 용도로 사용하는 데이터로 이를 활용한다면 각각 환경의 데이터를 수집할 수 있다.

하지만, 이를 사용하여 지형 데이터를 수집하는 방식은 느리고 복잡하다. 기본적으로 토지 피복도는 이미지 데이터이므로, 위도 경도를 Google Earth Engine에 전달하여 이미지를 얻어내고, 이 이미지의 픽셀값으로 라벨링 해야 한다. 이렇게 데이터를 수집하면 많은 양의 라벨링된 데이터를 수집할 수 있지만, 구글 어스 엔진 API 쿼리 제한 문제로 수집하는 시간이 많이 소모된다.



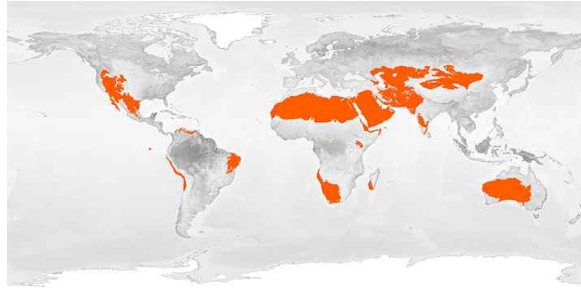


그림 8. Desert biomes (NASA)

그렇기에 위도 경도 탐색으로 데이터를 수집하는 방식 대신 2.5에서 살펴본 생물종이 서식할 확률을 나타내는 서식지 분포를 사용하여 위도와 경도를 얻어내어 생물종을 포함하는 생태계의 지형 데이터를 수집하였다. 즉, 사막 환경의 지형 데이터를 수집하고 싶으면, 낙타의 서식지 확률 분포를 구하여 이 분포에 해당하는 위도 경도로 지형 데이터를 수집하면 된다는 것이다.

이러한 수집 방식은 토지 피복도에서 라벨링된 환경보다 더 세부적인 환경으로 나눌 수 있다는 장점이 있다. 예를 들어 사막 생물 군계(biomes)는 그림 8과 같이 여러 지역에 존재한다. 모하비 사막, 사하라 사막, 고비 사막 등 각 사막에서만 서식하는 생물종이 있다면, 그 생물종이 서식하는 사막 지형을 하나의 라벨로 삼을 수 있다. 물론, 토지 피복도에서 사람의 손을 거쳐서 라벨을 추가할 수 있지만 번거로운 작업이다. 하지만 분포를 이용한 방식은 생물종 이름만 알고 있다면, 그 이후의 작업을 자동화할 수 있는 장점이 있다.

얻어낸 위도와 경도를 Google Earth Engine에서 제공하는 데이터셋 "COPERNICUS/S2\_SR"에서 위성사진을 수집할 수 있으며, 데이터 셋 "JAXA/ALOS/AW3D30/V3\_2"에서 DEM을 수집할 수 있다.

### 3.5 데이터 수집 및 전처리: 서식지 분포

데이터 수집하기 전에 생성할 환경에서 서식하는 생물종을 선정하였다.

- 모하비 사막에서 서식하는 조슈아 트리(Yucca Brevifolia)
- 유라시아와 북미의 산과 숲에서 서식하는 불곰(Ursus arctos)
- 고산지대에서 서식하는 겔라다 개코 원숭이(Theropithecus gelada)

2.5에서 살펴본 종분포모형을 적용하기 위해 수집해야 하는 최소 데이터는 종출현데이터와 기후데이터로 이들은 [12, 13, 14]에서 데이터를 수집한 것과 같이 GBIF(Global Biodiversity Information Facility)와 WorldClim에서 수집하였다.

GBIF는 생물 다양성에 대한 과학적인 자료를 수집하여 정리하는 국제 기구로 생물종의 출현 데이터를 제공해주며, WorldClim은 세계 기후 데이터를 제공해준다.



그림 9. WorldClim 1픽셀당 4.5km 샘플링 결과

이 프로젝트에서는 WorldClim 데이터는 1픽셀당 4.5km 해상도 데이터를 사용하였다. 이는 근접한 좌표데이터들을 제외해준다. 그렇기에 그림 9와 같이 일정한 간격 이상 떨어진 데이터들을 수집할 수 있다.

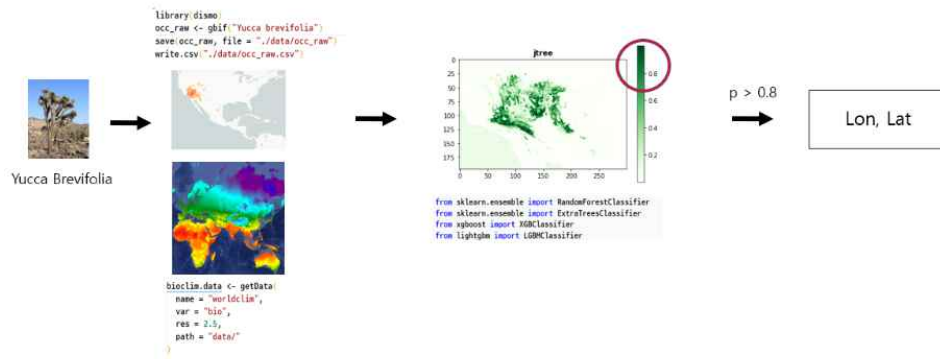


그림 10. Yucca Brevifolia 서식지의 경도 위도 수집과정

그림 10에서 보여지듯이 서식지 분포를 얻기 위한 과정은 다음과 같다.

1. GBIF 사이트 또는 R dismo 패키지의 gbif 함수로 출현 데이터 수집
2. WorldClim 사이트 또는 R dismo 패키지의 getData 함수로 기후데이터 수집
4. 지도학습에 필요한 R dismo 패키지에서 비출현 데이터 임의 생성(출현의 2배)
5. Python sklearn 라이브러리의 ensemble 모델 RandomForest 분류기 등을 사용하여 서식지 분포 계산[13]

생물종	출현/비출현 데이터 개수
불곰	39624
조슈아 트리	23979
겔라다개코원숭이	567

수집한 데이터의 개수는 위와 같으며, WorldClim에서 수집한 기후데이터를 경도 위도 좌표로 추출하였다. 추출한 변수들과 함께 출현, 비출현 정보를 클래스로 삼아 CSV 데이터 셋을 구축하였다. 학습은 겔라다개코원숭이 데이터의 개수가 적기 때문에 3개의 생물종 모두 K-fold 교차검증방법을 사용하였고, K는 [13]와 같이 5로 설정하였다. 결과는 다음과 같다.

분류기	5-fold 교차검증 정확도		
	불곰	조슈아 트리	겔라다개코원숭이
RandomForest	95.76(+/- 0.36)	98.93(+/- 0.17)	97.71(+/- 2.11)
ExtraTrees	95.71(+/- 0.32)	98.97(+/- 0.11)	98.24(+/- 2.49)
XGB	95.43(+/- 0.33)	98.86(+/- 0.10)	97.18(+/- 2.05)
LGBM	95.19(+/- 0.24)	98.86(+/- 0.08)	97.53(+/- 1.31)

이렇게 학습한 모델을 WorldClim 데이터에서 서식 확률을 추정하였고, 이들의 평균값을 사용하여, 그림 11에 해당하는 하나의 서식지 분포를 만들었다.

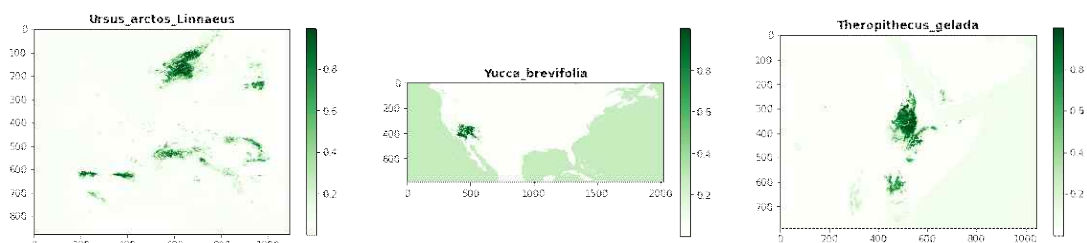


그림 11. 각 생물종의 서식지 분포



### 3.6 데이터 수집 및 전처리: 경도 위도

서식지 분포의 데이터 포맷인 tif 파일을 rasterio 라이브러리로 열고, 픽셀을 완전탐색하면서 0.8 확률 이상인 좌표를 추출하여 각각 생물종의 서식지 좌표 4500개를 수집할 수 있었다.

이렇게 얻은 좌표들은 다음과 같은 장점이 있다.

- 서식지 분포는 면적 데이터로, 출현 데이터가 적어도 좌표를 더 많이 수집 가능
- WorldClim의 픽셀당 4.5km 해상도로 인해 근접한 좌표 제외 가능

### 3.7 데이터 수집 및 전처리: 위성사진과 DEM

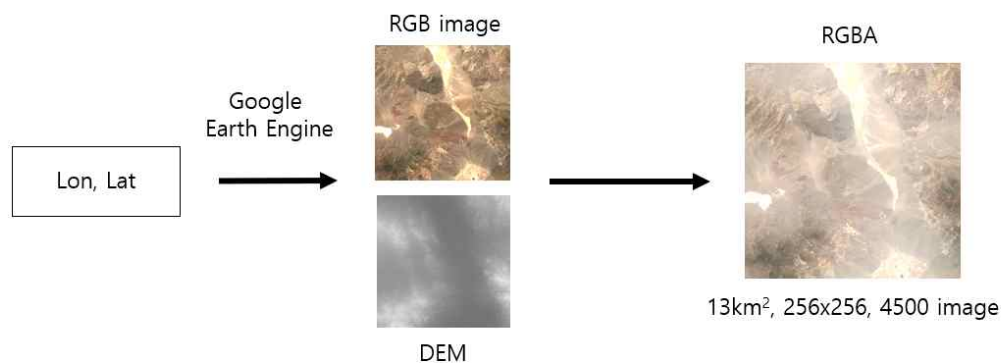


그림 12. Google Earth Engine으로 수집한 위성사진과 DEM을 하나로 합친 데이터

3.6에서 수집한 위도와 경도를 Google Earth Engine API 함수에 전달하면, 위성사진과 DEM을 수집할 수 있다. 여기서 품질이 저하될 수 있는 데이터는 위성사진으로, 위성사진은 촬영된 기간에 따라 변하는 특성이 있기 때문이다. 또한, 구름의 영향을 많이 받는 데이터이기도 하다.

Google Earth Engine의 위성사진 데이터 셋의 자료구조는 ImageCollection이다. 이 자료구조는 기간, 좌표, 채널 등에 필터를 적용하여 이미지들을 걸러낼 수 있으며, 남은 이미지들을 합성하여 하나의 이미지로 만들 수 있는 기능을 제공한다.

이 프로젝트에서는 ‘CLOUDY\_PIXEL\_PERCENTAGE’ 필터를 15로 설정하여 구름이 많은 이미지를 걸러내었고 남은 이미지에서 Median 연산으로 남은 이미지들의 픽셀값들의 중앙값을 취하여 이미지를 합성하였고, 결과는 그림 13과 같다.



그림 13. 겔라다개코원숭이 라벨 지형에서의 구름 필터와 Median 연산 결과

그림 13에서 보여지듯이, 구름이 없어진 것을 확인할 수 있다. 또한, 초록빛이 사라지는 현상을 확인할 수 있는데, 이는 Median 연산으로 인해 발생하는 것으로, 비교적 이 지역에서 촬영된 위성사진이 오른쪽과 같은 이미지가 더 많았음을 의미하며, 이 지역을 대표하는 위성 이미지라고 볼 수 있다.

이런 필터와 Median 연산을 생물 3종에 적용하여 그림 12와 같이 위성사진과 대응되는 DEM 이미지를 각각 4500개, 13km<sup>2</sup> 면적의 지형, 256x256 크기로 수집할 수 있었고, 2.2의 관련 연구들처럼 위성사진과 DEM 이미지를 하나의 RGBA로 합쳐서 데이터 셋을 구축하였다.

### 3.8 Conditional ProGANs

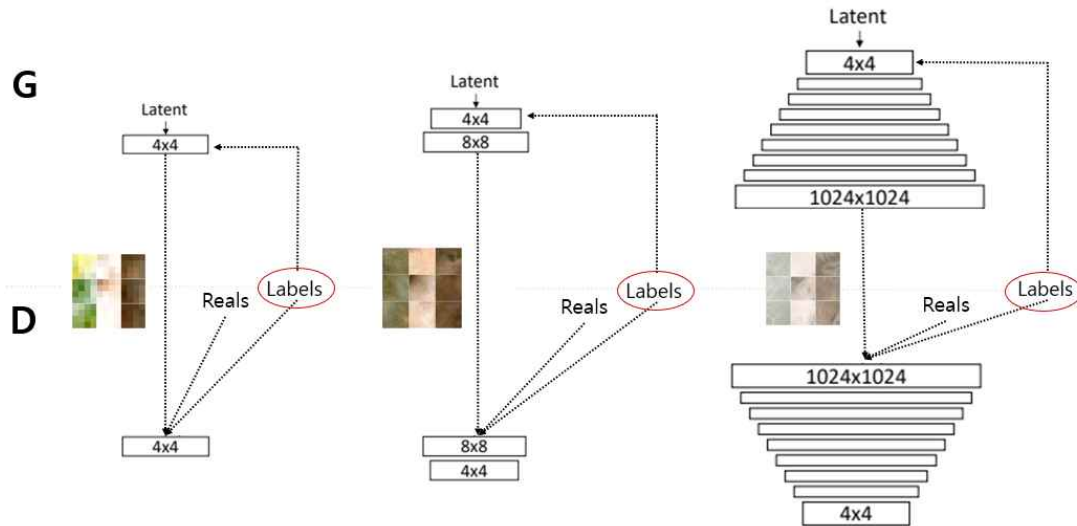


그림 14. Conditional ProGANs

이 프로젝트는 관련 연구[4]의 ProGANs를 기반으로 진행하였다. 구현된 ProGans 코드는 입출력이 3채널이었기 때문에 4채널로 수정하여 구축한 데이터 셋을 학습할 수 있게 하였다. 또한, 생물 3종을 라벨로 같이 학습시키기 위해 Conditional GANs[10]의 아이디어를 적용하였다. 이는 그림 14와 같이 라벨에 해당하는 y를 임베딩하여 입력과 연결하였다.

## IV. 학습 및 결과 시각화

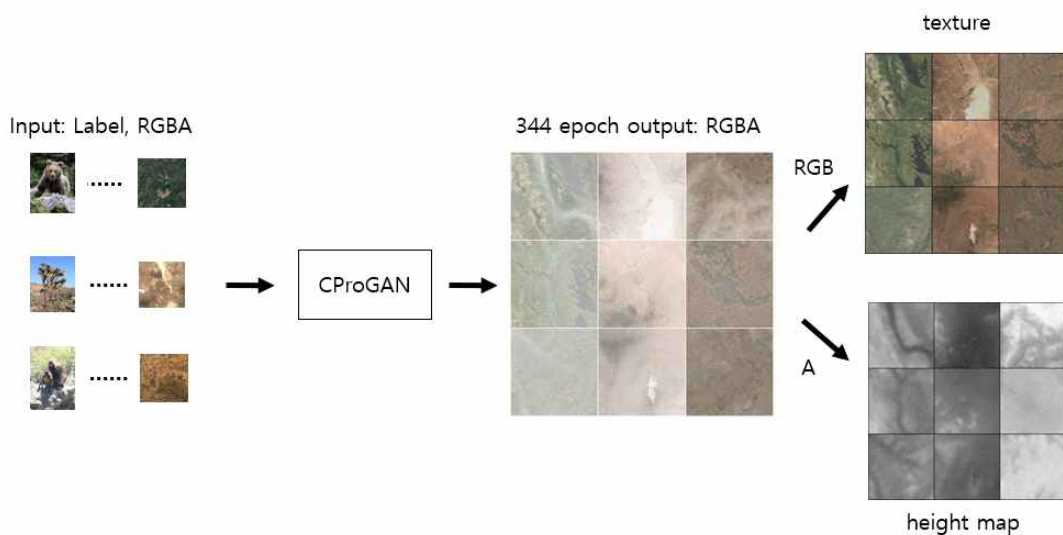


그림 15. 344 epoch 결과

3.7에서 구축한 데이터 셋과 3.8에서 수정한 모델로 344 epoch 학습시킨 결과는 그림 15와 같았다. 생성된 Texture는 실제 데이터와 비슷하였고, Height map은 숲을 의미하는 불곰 라벨에서 강이나 호수의 높이가 낮은 것을 확인할 수 있었다.

이렇게 학습된 모델의 결과물을 더 정확히 평가하기 위해 3D로 볼 수 있는 응용 프로그램을 구현하였다. 이는 이전 학기 프로젝트에서 구현한 모델 렌더러를 수정한 것으로, 지형 뷰어 기능 및 Redis를 통한 모델과의 통신 기능을 제공하여 지형을 렌더러에서 생성할 수 있다.

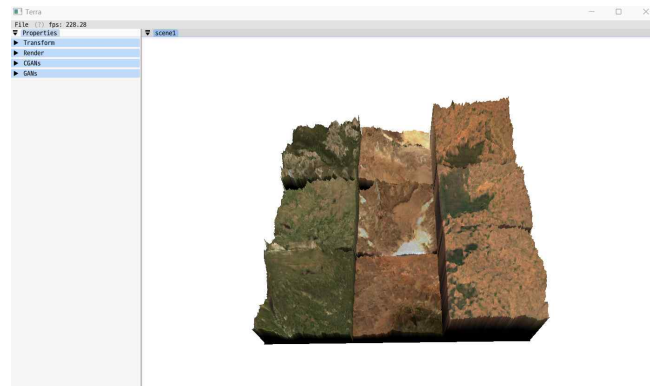


그림 16. 구현한 지형 뷰어

응용 프로그램에서 3D 지형으로 본 결과, 몇몇 특징이 보이는 결과들은 아래와 같았다.

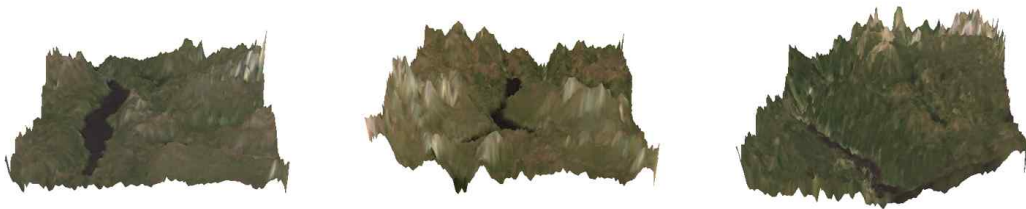


그림 17. 불곰 라벨

불곰 라벨에서 생성된 결과는 이미지에서 확인할 수 있었듯이 강 부분의 높이가 낮은 특징을 확인할 수 있다.



그림 18. 조슈아 트리 라벨

조슈아 트리 라벨에서 생성된 결과는 비교적 밝기가 높은 부분에서 높이가 낮았음을 확인할 수 있었다.



그림 19. 겔라다 개코 원숭이 라벨

겔라다 개코 원숭이 라벨에서 생성된 결과는 다른 2 종 같은 특징을 찾을 수는 없었지만, 비교적 어두운 부분에 경사가 형성되었음을 확인할 수 있었다.

## V. 결론 및 고찰

### 5.1 결론

3장에서 제시한 데이터 전처리를 통해 하나의 모델을 사용하는 방안의 결과는 그림 17, 18, 19에서 확인할 수 있었다. 즉, 색상과 높이 모두 RGBA 데이터로 학습시킬 수 있었다. 하지만, 높이는 아직 고르지 못하였다. 또한, 라벨로 원하는 지형 환경을 생성할 수 있었으며, 이를 위해 제시한 데이터 수집 방법 또한 코드로 모두 해결할 수 있었다.

### 5.2 고찰 및 향후 계획

이번 프로젝트에서 실패한 케이스는 종분포모델(서식지 분포) 부분이다. 이를 위해 동식물이 어디에 서식하는지 알아야 하며, 지도학습으로 분류했다는 한계가 있었다. 특히 지도학습을 사용하므로 임의로 비출현 데이터를 생성했어야 했고, 이러한 데이터들에 의해 바다에 서식할 확률이 표시되는 문제가 생겼다. 이러한 문제는 비지도 학습인 Maxent[14]로 해결할 수 있으므로 향후 Maxent를 사용하여 데이터 수집 과정을 하나의 코드로 정리할 계획이다.

다른 문제점은 라벨당 4500개의 데이터를 사용했다는 점, 관련 연구[3, 4]와 성능 비교를 하지 않았다는 점이다. 향후 하드웨어가 갖추어지면 다양한 라벨과 대량의 데이터를 가지고 관련 연구[3, 4]와 이 프로젝트를 함께 학습시켜보고 성능을 비교할 계획이다.

### 5.3 느낀점

데이터 수집부터 시작하여 인공지능 모델을 학습시켜보는 프로젝트는 이번이 처음이었기 때문에 GANs, CGANs, ProGANs, DCGANs 등 인공지능 모델에 대해 익숙해질 수 있는 좋은 경험이었다고 생각한다.

## 참 고 문 헌

- [1] 정윤경, 배병철, “게임에서의 절차적 생성기술(PCG)”, 정보과학회지 v.31 no.7, pp.16-25, 2013
- [2] Ryan J. Spick, Peter Cowling, James Alfred Walker, Procedural Generation using Spatial GANs for Region-Specific Learning of Elevation Data, IEEE, 2019
- [3] Christopher Beckham, Christopher Pal, “A step towards procedural terrain generation with GANs”, <https://arxiv.org/pdf/1707.03383.pdf>, 2017
- [4] Emmanouil Panagiotou, Eleni Charou, “Procedural 3D Terrain Generation using Generative Adversarial Networks”, <https://arxiv.org/pdf/2010.06411.pdf>, 2020
- [5] Md Kamal Uddin, AntonyLam, Hisato Fukuda, Yoshinori Kobayashi, Yoshinori Kuno, “Fusion in dissimilarity space for RGB-D person re-identification”, Array Volume 12, Dec 2021
- [6] Nels Numan, “Generative RGB-D Face Completion for Head-Mounted Display Removal”, IEEE VR, 2021
- [7] Yu Lin, Yigong Wang, Yifan Li, Yang Gao, Zhuoyi Wang, Latifur Khan, “Attention-Based Spatial Guidance for Image-to-Image Translation“, WACV, 2021
- [8] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, “Generative Adversarial Networks“, <https://arxiv.org/pdf/1406.2661.pdf>, 2014
- [9] Tero Karras, Timo Aila, Samuli Laine, Jaakko Lehtinen, “Progressive Growing of GANs for Improved Quality, Stability, and Variation“, <https://arxiv.org/pdf/1710.10196.pdf>, 2018
- [10] Mehdi Mirza, Simon Osindero, “Conditional Generative Adversarial Nets“, <https://arxiv.org/pdf/1411.1784.pdf>, 2014
- [11] 김은영, “생물다양성의 의미와 중요성”, 유네스코한국위원회, 한국사회와 지속가능발전 목표 15, pp.18
- [12] 권혁수, 류지은, 서창완, 김지연, 임동옥, 서민환, “종분포모형을 이용한 히어리 서식지의 분포 특성 연구“, 환경영향평가 제21권 제5호, pp.735-743, 2012
- [13] 권혁수, “종분포모형의 불확실성 확인을 위한 앙상블모형 적용”, 한국지형공간정보학회지, pp.47-52, Dec 2014
- [14] 김태근, 양두하, 조영호, 송교홍, 오장근, “Maxent 모델을 이용한 반달가슴곰의 서식지 분포변화 예측”, 생태와 환경 v.49 no.3, pp.197-207, 2016
- [15] Jialin Liu, Sam Snodgrass, Ahmed Khalifa, Sebastian Risi, Georgios N. Yannakakis, Julian Togelius, “Deep Learning for Procedural Content Generation“, 2020