

# GAN을 활용한 기상조건에 따른 하늘 이미지 생성

조규철\*, 조강현<sup>o</sup>

\*인하공업전문대학 컴퓨터정보공학과,

<sup>o</sup>인하공업전문대학 컴퓨터정보공학과

e-mail: kccho@inhac.ac.kr, jokh980324@naver.com

## Creating Sky Images according to Weather Conditions Using GAN

Cho Kyu Cheol\*, Jo Kang Hyeon<sup>o</sup>

\*Dept. of Computer Science, Inha Technical College,

<sup>o</sup>Dept. of Computer Science, Inha Technical College

### ● 요약 ●

현재 생성형 AI가 활발히 연구되고 있는 가운데, 대부분의 이미지 생성 AI는 프롬프트를 기반으로 한 Text-To-Image 방식을 주로 사용하고 있다. 하지만, 프롬프트 기반의 생성 AI는 실제 서비스에 도입하기 어려운 점이 많다. 여러 이미지 중, 하늘 이미지는 메타버스 등 가상 공간에서 매우 자주 사용되는 이미지 중 하나이면서 여러 입력값에 의해 이미지가 달라진다. 이 논문에서는 GAN을 활용해 기상 조건에 적합한 하늘 이미지를 생성하는 프로그램을 설계 및 구현한다.

**키워드:** 이미지 생성, GAN

## I. Introduction

현재 생성형 AI가 활발히 연구되고 있는 가운데, 많은 이미지 생성 AI는 프롬프트를 기반으로 한 Text-To-Image 방식을 주로 사용하고 있다. 이는 NovelAI사의 Stable Diffusion을 활용한 이미지 생성이 화제가 된 이후, 소스코드 유출로 인해 해당 기술을 기반으로 한 WebUI 등 손쉽게 프롬프트를 이미지로 변환하고, AI를 학습시킬 수 있는 도구가 빠르게 확산된 영향으로 보인다.

Text-To-Image 방식은 일반 사용자들이 쉽게 접근할 수 있다는 점에서 사용자들에게 인기를 끌었지만, 단어가 아닌 수치를 입력할 수 없다는 한계점이 존재하여 디지털 트윈 등을 위한 실 사용용으로는 최고의 방법이라고 할 수 없을 것이다.

하늘 이미지는 외부 상황이 보이는 공간에서의 디지털 트윈 가상공간 구현에 항상 사용되는 부분이지만, 대부분의 경우 맑음, 흐림, 비 정도만 구현하는 경우가 대부분이다. 따라서 본 논문에서는 가중치로 조정될 수 있는 생성형 AI 제작을 설계하고 구현하며, 그 대상으로 하늘 이미지를 선택하였다.

본 연구는 2장에서, AI 학습을 위한 데이터를 수집하는 센서 장치의 설계하고 구현하며, 3장에서는 해당 데이터의 전처리와 보정 작업, 4장에서는 AI의 레이어 구성에 대해 다룬다.

## II. Data Collection

### 2.1. Experiment Data

현재 하늘을 찍은 이미지 데이터와 그와 연관된 기상 데이터를 찾는 것은 매우 어렵다. 따라서 데이터를 직접 수집해야 했으며, 현재 상황을 고려하여 카메라를 통한 이미지, 온도도, 조도를 수집할 데이터로 결정하였다.

### 2.2. Data Collection Method

데이터 수집을 목적으로 설치할 기기는, 카메라를 사용하며, 방수를 위해 밀봉하였다. 그래서 무선 또는 유선으로 프로그램을 수정할 수 있어야 했다. 따라서, 무선 네트워크를 통해 원격 제어가 가능하며, 내장된 프로그램 수정을 위해 별도의 유선 연결이 불필요한 라즈베리파이를 통해 데이터 수집을 진행하였다.

또한, 센서 데이터 수집을 위해 이두이노를 라즈베리파이와 연결하였다. 이는 아날로그 데이터 수집 등 라즈베리파이로 직접 연결할 수 없는 측정 장비도 연결할 수 있었으며, 라즈베리파이를 통해 이두이노의 프로그램 또한 수정할 수 있었다.

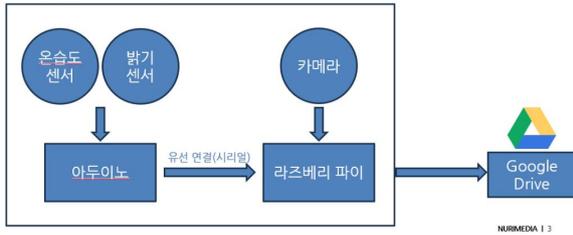


Fig. 1. Data Collection Devices Diagram

그림1의 데이터 수집을 진행하기 위해 구현된 실제 수집 장치의 모습은 그림 2와 같다.



Fig. 2. Data Collection Devices

해당 장치는 Google Drive API[1]를 사용하여, 이미지에 라벨을 붙여 업로드되며, 데이터는 학습 이전에 일괄적으로 다운로드 되고 전처리한다.

### III. Image Preprocessing



Fig. 3. Collected Image

그림 3은 그림 2의 장치를 통해 생성된 사진 데이터이다. 사진의 가장자리에 색상 왜곡이 보이며, 하단에는 지상 구조물이 함께 찍혀 있다. 사진 데이터에서 필요한 사진 데이터로 가공하기 위해 그림 4와 같은 전처리 과정을 거쳤다.

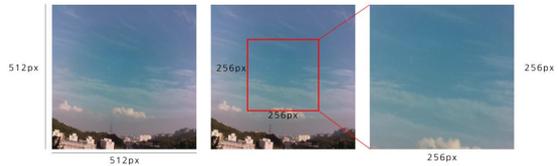


Fig. 4. Image preprocessing

사진은 가로와 세로 각각 512px의 원본 이미지에서, 중앙영역에 대해 256px 크기의 정사각형 부분만을 잘라 가공하여 그림 5와 같이 사용하였다.

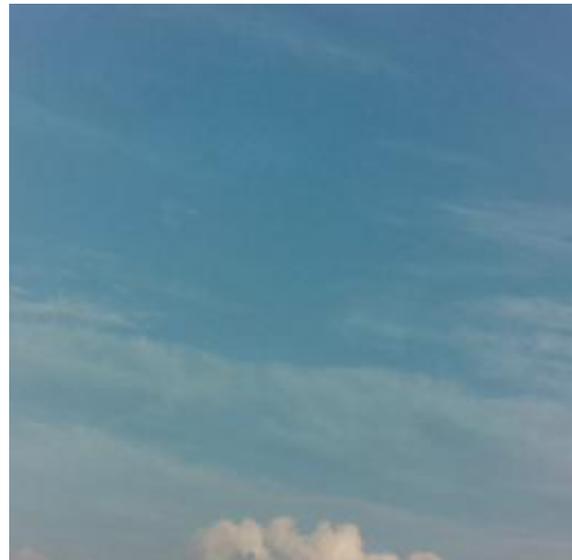


Fig. 5. Image preprocessing result

그 결과 전처리를 통해 그림 5와 같은 하늘 이미지를 획득할 수 있었으며, 조금 더 선명한 하늘 이미지를 얻기 위해 일몰 이후의 야간 시간대는 이미지를 데이터에서 제외하였다.

### IV. AI Structure & Implementation

이미지 생성은 GAN모델을 Pytorch를 통해 구현되었다. 이미지 데이터에 여러 개의 라벨이 붙어 있어, Dataset 클래스를 상속받은 CustomDataset을 구현하였으며, pytorch의 DataLoader를 통해 과 일을 적재하였다.

#### 4.1. Generative Network(G)

생성망은 입력값으로 년, 월, 일, 시, 분, 초, 온도, 습도, 밝기 총 9개의 입력값과 247개의 랜덤 노이즈를 입력받으며, 출력으로

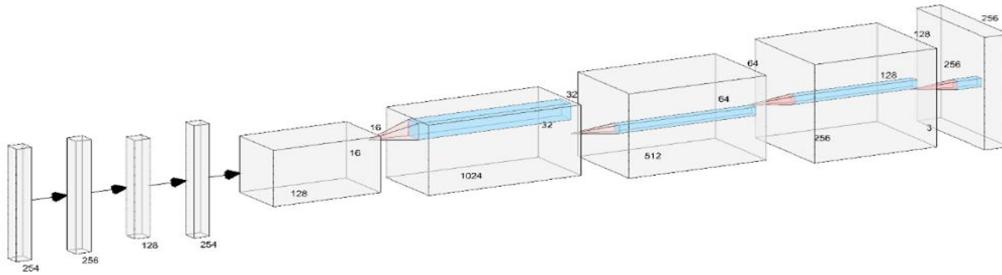


Fig. 7. Generator structure

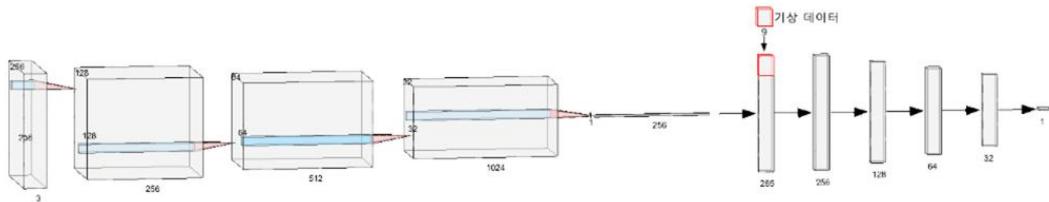


Fig. 8. Discriminator structure

RGB 컬러 이미지를 생성하도록 학습시킨다.

생성망은 그림 7과 같이 4개의 Linear 레이어와 5개의 Conv레이어를 가지고 있다. 각 레이어 사이에 BatchNorm을 적용하였고, 활성화 함수는 마지막 레이어를 제외한 레이어에서는 ReLU, 마지막 레이어에서는 Tanh를 사용하였다.

학습을 진행한 결과, 그림 9와 같은 Loss 그래프를 얻어낼 수 있었으며, 오후 2시를 기준으로 이미지를 생성하였을 경우, 가로와 세로축을 각각 온도와 습도로 하여 그림 10과 같은 이미지를 얻어낼 수 있었다.

#### 4.2. Discriminant Network(D)

판별망은 가로세로 각각 256px의 RGB 컬러 이미지를 입력받으며, 출력으로 입력이 생성된 이미지이면 0, 원본 이미지라면 1으로 출력하도록 학습시킨다.

판별망은 그림 8과 같이 4개의 Conv레이어를 먼저 거친다. Conv레이어의 활성화 함수는 LeakyReLU이며, BatchNorm을 사용하였다. 결과로 256개의 값을 출력하며, 여기에 생성망에서 사용했던 9개 변수를 더해 265개의 값을 4층의 Linear레이어에 통과시킨다. Linear레이어의 활성화 함수는 ReLU이며, 마지막 레이어에서는 Sigmoid 활성화 함수를 사용하였다.

#### 4.3. Result of Training

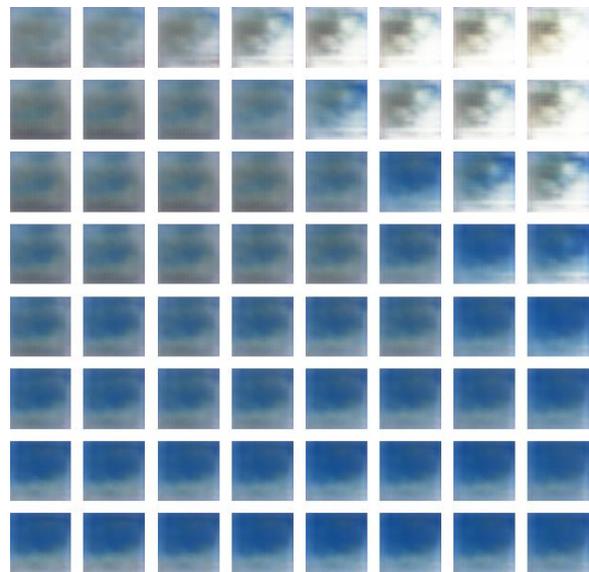


Fig. 10. Generated image

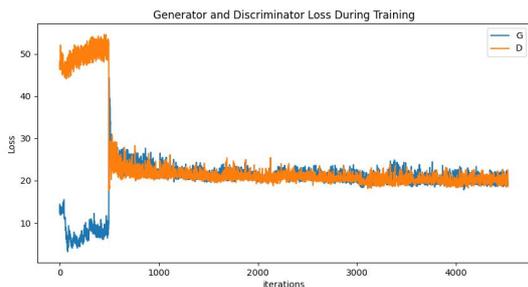


Fig. 9. Loss graph

#### V. Conclusions

기상 공간을 위해 필요한 리소스의 수나 품질의 요구 수준은 점점 늘어나고 있으며, AI는 이런 수요에 적합한 도구로 응용할 수 있는 부분은 점점 늘어나게 될 것이다. 본 논문에서는 GAN을 활용하여 하늘 이미지를 생성하는 AI를 구현하였다. 낮은 해상도, 이전 이미지를 통한 연속적인 이미지 생성이 불가능하다는 한계점이 존재하지만, 추가 연구를 통해 개선될 수 있을 것이다.

## REFERENCES

- [1] Google, Google Drive API (<https://developers.google.com/drive/api/reference/rest/v3?hl=ko>)
- [2] Pytorch Foundation, DCGAN TUTORIAL ([https://pytorch.org/tutorials/beginner/dcgan\\_faces\\_tutorial.html](https://pytorch.org/tutorials/beginner/dcgan_faces_tutorial.html))