

Cloud Semantic Segmentation from Satellite Images

서이석, 유도현, 추승민

중앙대학교 응용통계학과

요약

지구 환경의 변화를 파악하는데 중요한 환경요소인 구름은 태양광 반사 작용, 지표 온도 상승 문제와 관련되어 있고 위치와 두께, 내부 구성에 따라 각기 다른 기상 현상을 일으키기 때문에 중요한 분석 대상으로 꼽힌다. 위성영상은 지구의 대기권을 촬영하여 광범위한 지역에서 많은 종류의 구름을 기록할 수 있어 구름을 분석하기 위한 자료로 널리 활용되고 있다. 그러나 위성영상에서는 구름 간의 경계가 모호해 분류에 이용되는 자동화되고 체계적인 알고리즘을 구성하기 어렵고, 연구자들의 수작업으로는 데이터를 확보하는데 많은 시간이 소모되는 문제가 있었다. 본 연구에서는 딥러닝 기반의 의미론적 분할 모델인 U-Net, DeeplabV3+, HRNetV2를 활용해 해당 분류 작업을 자동화함과 동시에 Convex Hull 기법으로 구름 간의 합리적인 경계선을 도출하는 실험을 진행했다. 학습자료로 Kaggle의 데이터 경진대회 중 “Understanding Clouds from Satellite Images”에서 제공한 위성 이미지데이터를 활용하였으며 딥러닝을 통한 분할이 구름의 형태별 구분을 효과적으로 수행할 수 있음을 밝혔다. 또 가장 분할 성능이 좋은 모델은 HRNetV2이었으며 Convex Hull의 윤곽선 확장 범위는 미미하지만 확실히 합리적이고 매끄러운 윤곽선을 도출해냈다.

주요 용어: 구름, 의미론적 분할, 위성영상, Convex Hull

1. 서론

지구 온난화에 따른 기후 변화가 세계 곳곳에 영향을 미침에 따라 지구 환경에 대한 관심이 증가하고 있다. 특히 지표 온도의 변화에 직접적으로 영향을 받는 구름은 강수, 태양광 흡수 및 반사 작용에 핵심적인 환경 요소로서 가장 중요한 분석 대상 중 하나로 꼽히고 있다. 구름은 형태와 크기에 따라 지구에 미치는 영향이 각각 다르기 때문에 연구를 진행하기에 앞서 이러한 분류 과정이 정밀하게 시행되어야 한다. 위성영상은 광범위한 지역에 대한 주기적인 탐사 정보를 제공하는 시각 자료로 많은 종류의 구름들이 하나의 위성영상에 동시에 나타나있어 구름 연구에 널리 이용되고 있다. 현재 숙련된 연구진들이 이러한 구름들을 분류하는 작업을 맡고 있으나, 구름 형태 간의 경계가 상당히 모호해 분류에 필요한 체계적인 알고리즘을 구성하기 어렵다. 또한 분류를 담당하는 고급 인력의 부족으로 인해 연구에 필요한 방대한 양의 데이터를 쌓는데 많은 시간이 소모되고, 다중적인 분광 정보가 주어지지 않을 시 정확도가 떨어지는 문제점이 있었다. 우리는 이미지 속 객체 분류 및 분할에 뛰어난 성능을 보이는 합성곱 신경망(CNN)을 기반으로 한 다양한 딥러닝 모델을 구축하여 해당 문제점들을 극복하고 구름 분류에 가장 성능이 뛰어난 모델을 찾고자 한다. 그리고 구름 간의 적절한 경계 또한 찾아내기 위해 우리는 컴퓨터 비전 분야에서의 여러 과제 중 의미론적 분할로 구름 분류 문제에 접근했다.

본 논문은 총 5장으로 구성되어 있다. 2장에서는 선행 연구로 기존의 의미론적 분할 모델과 그것이 구름 분류에 적용된 사례에 대해 설명하고 3장에서는 제안하는 모델의 구성 및 특징을 설명한다. 4장은 실험에 사용한 데이터에 관한 설명과 처리 과정 및 결과를 제시한다. 5장은 이 논문의 결론을 서술한다.

2. 선행 연구

의미론적 분할은 실제 사람이 물체를 인식하는 방법처럼 객체를 특성에 따라 분류함과 동시에 객체의 위치 정보를 정밀하게 표현하는 기술로 컴퓨터 비전 분야에서 가장 중요한 과제 중 하나이다. 의미론적 분할의 핵심은 객체 분류를 픽셀 단위로 진행한다는 것이다. 이를 위해 의미론적 분할 모델은 입력에 사용되는 원본 이미지를 여러 층으로 구성된 합성곱 신경망에 통과시켜 높이와 너비가 사진과 동일한 레이블 맵을 출력한다. Fully Convolutional Network(FCN)은 이와 같은 방식을 도입한 대표적인 초기 분할 모델이다. FCN은 우선 이미지 분류 모델인 VGG16 네트워크 뒷단의 완전연결 계층(Fully Connected Layer)을 1x1 합성곱 계층으로 대체하여 객체의 위치 정보를 보존하면서 분류를 위해 특징 맵의 채널 수를 분할하고자 하는 객체 종류에 맞게 조정하고자 했다. 그리고 계층을 통과하면서 작아진 특징 맵의 크기를 업샘플링을 통해 원본 이미지와 동일한 크기로 조정하여 픽셀 단위의 예측값이 특징 맵에 포함되도록 한다(Jonathan Long et al., 2015).

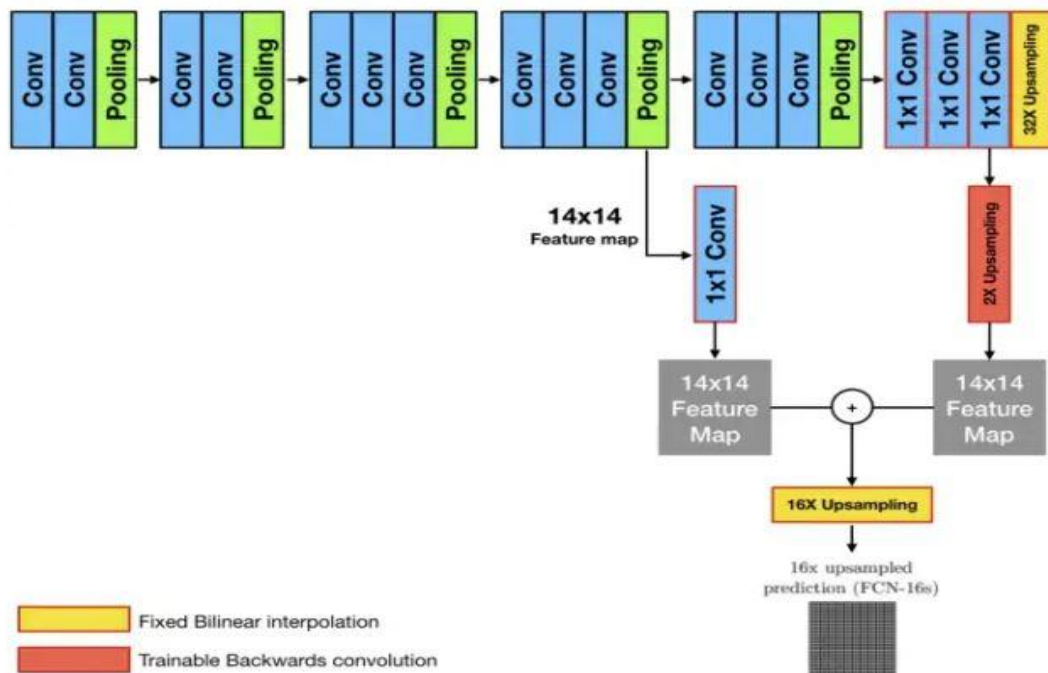


그림 1. FCN-16s 구조

이 FCN을 기반으로 많은 분할 모델이 탄생하였다. 대표적으로는 같은 해에 의학 분야에서 제안된 U-Net이 등장하였고, 합성곱층이 정보를 읽어들이는 수용영역의 크기를 다양화하는 Pyramid Pooling 기법을 추가한 PSPNet, Deeplab 등이 의미론적 분할의 성능을 끌어올렸다. 이후에도 고해상도의 이미지에서 발견된 원본 이미지의 손실을 최소화하는 HRNetV2 등이 개발되어 높은 성능을 기록하는 등 의미론적 분할 기술은 빠르게 발전하고 있는 추세이다.

이러한 의미론적 분할을 구름이라는 부정형 객체에 적용한 사례는 크게 2가지 목적으로 구분할 수 있다. 첫 번째는 위성을 통한 건물, 차량 등의 탐지에 방해가 되는 구름이 많이 포함된 이미지를 걸러냄으로써 데이터의 품질을 향상시키기 위한 목적이다. 한 연구에서는 이미지 분할 모델 중 하나인 DeeplabV3+ 를 이용해 구름을 분할하고자 했다(성선경 등, 2020). 해당 연구는 구름이 포함된 위성사진 학습자료를 최대한

확보하기 위해 근적외선 파장까지 촬영이 가능한 Landsat 8 위성 데이터를 통해 모델을 선학습시킨 후, 가시광선 파장으로 한정된 PlanetScope 위성자료로 모델을 전이학습시키는 방식으로 진행되었으며 그 결과 정밀도는 전이학습 전과 비교해 약 49% 상승했으나, 재현율은 24% 감소했다. 이는 구름이라는 객체에 한해 사전에 훈련된 가중치를 모델에 불러오는 작업이 항상 좋지만은 않음을 시사하고 있다. 다른 논문에서는 U-Net 모델에 Dilated Convolution을 적용하여 커널의 수용영역을 확장시키고 손실함수로 다이스 손실을 사용해 모델을 학습한 결과를 소개했다(김보람 등, 2020). 저자는 이미지 분할에 주로 사용되는 손실함수인 교차 엔트로피는 픽셀 단위로 손실을 계산하여 전체 영역의 정보를 학습 과정에서 잘 활용하지 못하기 때문에 이를 보완하고자 다이스 손실을 사용했음을 논문에서 밝히고 있다. 얇은 구름은 잘 분류하지 못했지만 짙은 구름 영역에서 예측 성능이 좋았다.

두 번째는 기상학 연구를 위해 구름을 유형별로 분류하여 각각을 분석하기 위한 목적이다. 한 논문에서는 구름 분할을 위해 자체적으로 개발한 CLP-CNN 모델을 이용해 약 84.4%의 픽셀별 분류 정확도를 기록했다(Jiang, Y et al., 2022). CLP-CNN은 기존 U-Net에서 합성곱 계층을 잔차 블록으로 교체하여 특징을 추출하고 Attention 기법을 활용해 네트워크 스스로 중요한 영역에 더 높은 가중치를 할당할 수 있도록 했다. 서로 다른 크기의 특징 맵을 통합하는 ASPP를 이용하였으며, 다운샘플링 과정에서 발생하는 정보 손실을 방지하기 위해 풀링 계층 대신 보폭이 2인 합성곱 계층을 이용했다. 총 9종류의 구름에 대해 비교 모델인 U-Net++ 보다 평균 5%의 정확도 향상을 기록했다.

3. 모델 제안

본 연구는 위성영상에서 구름을 분류하는 선행 연구에서 좋은 성능을 보인 U-Net과 Deeplab, 우리가 새롭게 적용할 모델인 HRNetV2를 기반으로 다양한 백본 네트워크를 결합시켜 구름 분할에 최적의 성능을 보이는 모델을 경험적 접근법을 통해 탐색한다.

U-Net은 인코더와 디코더 구조가 단순하기 때문에 다양한 형태로 쉽게 변형시킬 수 있다는 장점이 있고, 특히 이미지 속 객체의 특징을 학습하는 구간인 인코더를 보다 더 세밀한 학습이 가능한 신경망 구조로 대체하는 방법이 많이 사용되고 있다. 이번 연구 또한 인코더 부분을 이미지 분류에 사용되는 딥러닝 모델의 백본 네트워크로 교체한 뒤 디코더에 Skip Connection을 통해 정보를 전달하는 U-Net의 방식을 활용하여 모델을 구축하였다. 이용된 딥러닝 분류 모델로는 기존 U-Net 인코더에 더 많은 합성곱층을 추가한 VGG19, 잔차 블록을 활용한 ResNet18과 ResNet50, Depthwise Convolution을 활용한 MobileNetV2, EfficientNet-B1이 있다. 그러나 VGG와 ResNet의 경우 모델의 깊이를 증가시켜 성능을 높이는 구조이기 때문에 백본이 깊어질수록 연산량이 급격히 증가하게 되어 하드웨어의 메모리 부족 문제가 발생할 수 있다. 하지만 MobileNet은 Depthwise Convolution을 통해 특징 맵의 각 채널들을 여러 공간에 병렬적으로 분리시킨 후 동시에 연산을 수행하여 연산량을 획기적으로 줄임으로써 이러한 문제에 직면할 가능성을 낮췄다. 또한 EfficientNet은 성능 향상을 위해 입력 이미지의 해상도, 너비, 모델의 깊이에 대한 최적의 조합을 찾아내는 알고리즘을 도입해 파라미터 수의 증가를 억제하고 연산량 또한 낮게 유지한다. 현재 연구팀의 연구 환경이 Google Colab, 일반 데스크탑 컴퓨터로 상당히 열악하기 때문에 모델의 성능에 집중함과 동시에 이런 연산량 문제를 고려했다.

Deeplab 모델은 필터 사이에 빈 공간을 만들어 특징을 추출하는 Dilated Convolution이 중심이 되는 딥러닝 분할 모델이다. 이 기법은 합성곱 계산 시 수용 영역의 크기를 증가시키기 때문에 정보 손실이 줄어들게 되어 디테일한 특징을 학습해야 하는 분할 과제에 적합하다. 특히 2장에서 제시된 구름 분할을 목표로 하는 선행 연구들에서 공통적으로 Dilated Convolution 또는 이를 포함하는 Deeplab 모델이 사용되었으며 좋은 성과를 거두었다. Deeplab은 처음 소개된 이후 다양한 크기의 문맥 정보를 잘 학습하기 위해 Atrous Spatial Pyramid Pooling (ASPP)를 활용한 DeeplabV2나 Depthwise Separable Convolution과 Dilated Convolution을 결합한 Atrous Separable Convolution의 활용을 제안한

DeeplabV3+와 같이 발전되어왔다. 때문에 본 연구에서도 Deeplab을 기반으로 하는 분할 모델을 이용하였다.

HRNetV2는 분류 모델인 HRNet에서 뒷단의 완전연결계층을 1x1 합성곱층으로 교체하여 분할 과제를 수행할 수 있게 한 모델이다. 일반적인 분할 모델은 고해상도의 원본 이미지를 특징 추출 과정에서 저해상도로 낮추고, 추출된 특징 맵을 다시 고해상도로 변환하는 과정에서 정보 손실이 발생하지만, HRNetV2의 백본은 이미지를 저해상도로 낮출 때 기존의 고해상도 이미지를 삭제하지 않고 1x1 합성곱층 전까지 유지시키는 구조를 통해 정보 손실을 최소화한다. 분할 성능 향상을 위해 모델을 더 깊게 하거나 구조가 더 복잡해지지 않기 때문에 연산량이 크게 늘지 않는 장점 또한 보유하고 있다.

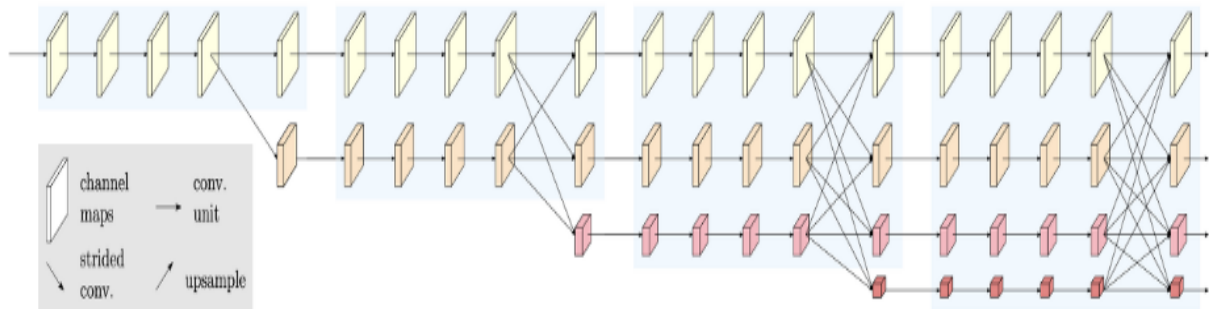


그림 2. HRNetV2 Backbone 구조

이렇게 선정된 분할 모델들은 이진 교차 엔트로피(Binary Cross Entropy)와 다이스 손실(Dice loss function)이 혼합된 형태의 손실함수를 사용하여 훈련시킨다. 교차 엔트로피는 분할 모델에 일반적으로 이용되는 손실함수이지만 이미지 전체 영역의 정보를 잘 활용하지 못하기 때문에 클래스 불균형이 발생할 경우 학습이 제대로 이루어지지 않는다. 다이스 손실을 사용하면 이 문제를 보완할 수 있으나, 픽셀별 분류 정확도는 일반적으로 교차 엔트로피가 더 뛰어나므로 데이터에서 클래스 불균형이 극명하게 나타나는지를 먼저 분석해야 한다. 그러나 클래스 불균형 여부에 대한 판단이 애매할 경우 다음과 같이 두 손실함수를 혼합한 형태를 이용할 수 있다.

$$BCE = -\frac{1}{N} \sum_{i=1}^N y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)$$

$$DICE = 1 - 2 \frac{\sum_{i=1}^N y_i \hat{y}_i}{\sum_{i=1}^N y_i + \sum_{i=1}^N \hat{y}_i}$$

$$LOSS = BCE + DICE$$

여기서 y_i 는 i 번째 픽셀의 실제 클래스 값을, \hat{y}_i 는 i 번째 픽셀의 예측된 클래스 값을 의미한다. 클래스에 해당하면 1, 배경에 해당하면 0으로 환산된다. 명시되어있지 않은 한, 손실함수에 대해 단순 덧셈으로 계산했기 때문에 손실함수의 최대값은 2이다.

4. 실험

우리는 Kaggle의 대회 중 “Understanding Clouds from Satellite Images”에서 제공하는 데이터셋을 이용해 실험을 진행하였다. 모델의 성능을 측정할 때는 Kaggle에서 제시하는 측정 방식인 다이스 계수를 활용하였다. 또한 Kaggle의 규칙 상 최종 이미지는 원본 사이즈(높이 1400, 너비 2100)의 1/4에 해당하는 크기(350, 525)로 테스트용 이미지를 축소한 후 예측을 진행해야 했다.

4.1. 데이터 설명

사용한 데이터셋은 구름이 촬영되어있는 5546장의 훈련용 이미지와 3698장의 테스트용 이미지로 구성되어 있으며, 분류해야 할 구름 패턴은 총 4가지로 구름의 형태 차이에 따라 Fish, Flower, Gravel, Sugar라는 이름이 붙었다. 해당 구름 영역에 대한 각 레이블은 Max-Planck Institute 소속 과학자들의 주관에 의해 생성되었다(총 68명). 구름의 패턴과 패턴 사이를 구분하는 기준이 불분명하여 연구자들이 하나의 구름 패턴이 차지하는 영역을 서로 다르게 판단할 가능성이 있어, 이 경우 최종적인 레이블은 각 연구자가 분류한 영역의 합집합으로 정해졌다.

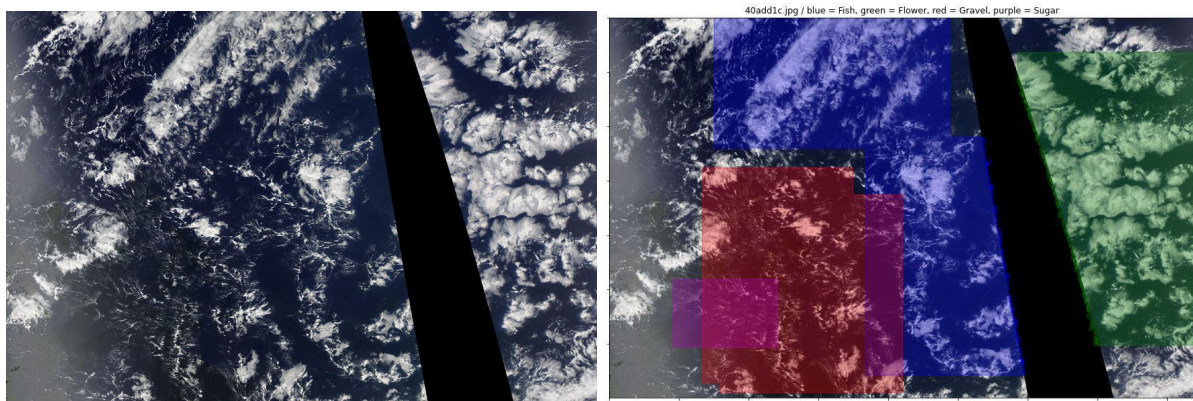


그림 3. 원본 이미지(좌) 및 Ground-truth 영역 레이블(우) (파랑 : Fish, 초록 : Flower, 빨강 : Gravel, 보라 : Sugar)

따라서 해당 이미지처럼 구름의 패턴 영역이 서로 겹치는 경우가 있으며, 영역의 윤곽선 또한 정밀하지 않은 사각형 형태이므로 주어진 레이블이 구름 패턴에 대한 정확한 정보를 표현하고 있지 않음을 감안해야 한다. 또한 이미지들이 극궤도 위성에서 촬영되었기 때문에 지구 자전의 영향을 받아 촬영이 불가능한 지역이 존재하게 되어, 해당 부분은 이미지에서 검은색으로 표현되었다.

4.2. 실험 준비

모델을 학습하기에 앞서 복잡한 전처리 과정을 거쳤다. 우선 레이블이 [그림 3]처럼 이미지가 아닌 Run-length encoding을 거쳐 문자열 형태로 압축되어 있어 이를 Numpy array 형식으로 재구성하였고, 이미지 또한 같은 형식으로 변환하였다. 이 과정에서 이미지 해상도를 높이 384, 너비 576으로 크게 감소시켰는데, 원본 크기의 이미지로는 모델을 변경하고 간소화시켜도 메모리 부족 문제로 학습이 불가능했기 때문이다. 견고한 모델 학습을 위해 이미지마다 0.5의 확률로 Augmentation을 진행하였으며 좌우 방향 뒤집기, 상하 방향 뒤집기, 왜곡 방법을 사용했다. 또한 구름 영역이 서로 겹쳐있는 경우가 존재하기 때문에 모델의 출력 채널 수를 구름 패턴 종류와 동일한 4로 설정하여 각 채널마다 패턴과 배경 간의 이진 분류 결과를 담을 수 있도록 했다.

모델의 가중치 최적화 알고리즘은 Adam을 사용했으며 배치 사이즈는 32를 일괄적으로 적용했다. 학습 과정에서 과적합을 방지하기 위해 Early Stopping을 적용했고 학습이 더더질 때마다 기존 학습률 값을 1/2씩 감소시키는 방법을 사용하여 모델이 최소의 손실값에 도달할 수 있도록 했다. 실험에 사용된 모든 모델은 우리의 연구 환경에서 제대로 작동하는지 먼저 확인한 뒤 커널의 자원 수, 합성곱층 깊이 등의 초매개변수를 튜닝하였다. 그러나 메모리 부족 문제로 인해 실제 연구에서는 대부분의 파라미터 값을 우리가 원하는 수준보다 낮게 유지해야만 했다.

학습 이후엔 주어진 구름 영역 레이블이 합집합이기 때문에 실제 영역보다 크게 설정되어 있는 문제를 고려하여, 영역의 윤곽선을 확장하는 기법 중 하나인 Convex Hull을 사용해 예측 결과를 후처리하였다. Convex Hull은 2차원 좌표평면 상에서 지정된 모든 점을 포함하는 볼록한 껍질을 생성하는 기법으로, 이미지 분할에 적용하면 객체 영역으로 예측된 모든 픽셀을 포함하는 껍질 형태의 윤곽선을 정의할 수 있다.

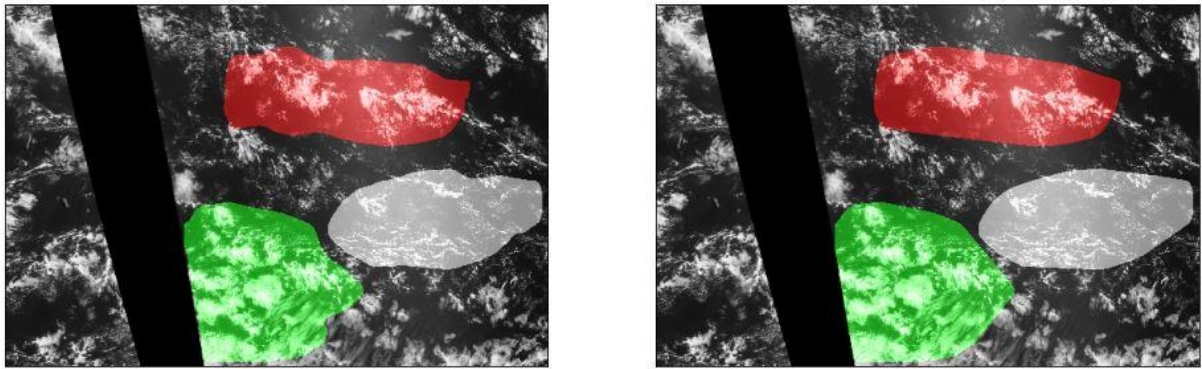


그림 4. 모델의 예측(좌)과 Convex Hull 적용 후(우)

위 사진처럼 Convex Hull은 모델의 예측에 기반하여 영역을 확장시킬 수 있으며 윤곽선의 형태도 매끄럽게 조정할 수 있다.

또한 원본 이미지 사이즈 기준으로 5만 픽셀 미만의 영역을 차지하는 패턴이 훈련 데이터 전체에서 10개에 불과했기 때문에, 우리는 예측 성능 향상을 위해 학습된 모델이 예측한 구름 패턴의 영역이 일정 픽셀 이하일 경우 배경으로 처리하도록 했다. 기준이 되는 픽셀 값은 초매개변수이므로 튜닝 과정을 거쳐 15000으로 정해졌다. 이는 예측이 원본 사이즈의 1/4 크기에서 진행되어야 하는 Kaggle 대회 규칙을 반영한 것이다.

4.3. 결과

세 가지 모델 U-Net, DeeplabV3+, HRNetV2의 성능 비교를 진행하였다. Table 1은 훈련 데이터에서 U-Net, DeeplabV3+, HRNetV2의 학습을 비교한 결과 표이다. Table 2는 테스트용 데이터에서 모델이 예측한 결과와 후처리 과정을 추가한 결과를 비교한 표이다. 위에서 설명했듯이 후처리 과정에는 예측된 구름 패턴이 일정 픽셀 이하의 영역을 가질 경우 해당 예측을 제외하는 Min_size Filter와 윤곽선을 개선 및 확장하는 Convex Hull이 포함되어 있다. Table 1에서 훈련 성능이 좋았던 5개 모델을 선정하여 결과를 출력했다. Table 3는 선정된 5개의 모델을 여러 조합으로 앙상블한 결과를 제시한 표이다. 최고의 Kaggle 평가 점수를 얻기 위해 진행하였으며 소프트 보팅 방식의 앙상블을 적용했다.

Table 1. Training results

Model	Backbone	Train loss (0~2 scale)	Train dice coef.	Validation loss (0~2 scale)	Validation dice coef.
U-Net	(default)	0.9704	0.3922	1.0703	0.4085
	VGG19	0.7625	0.5352	0.7711	0.5270
	ResNet18	0.8013	0.5130	1.8592	0.2065
	ResNet50	1.0118	0.3777	1.5210	0.2129
	MobileNetV2	0.6235	0.6296	0.9376	0.4692
	EfficientNet-B1	0.6738	0.5962	0.7370	0.5551
DeeplabV3+	ResNet50	0.8047	0.5079	0.9670	0.4652
HRNetV2	HRNet	0.7686	0.5310	0.7548	0.5397

Table 1을 보면 백본으로 EfficientNet-B1을 사용한 U-Net 모델이 Validation set에서 가장 적은 손실값을 기록했으며 다이스 계수 값 또한 가장 좋았다. 그 외에 VGG19를 사용한 U-Net, HRNetV2이 높은 성능을 나타냈다. 깊은 모델 혹은 디테일한 정보의 손실을 최소화하는 방식을 도입한 모델일수록 학습 결과가 전반적으로 좋아진다는 사실을 확인할 수 있다. 그러나 ResNet 기반의 백본을 도입한 U-Net 모델은 학습의 첫 에폭부터 과적합이 발생하여 성능이 매우 낮았다. 유독 ResNet 백본에서 이런 결과가 나타난 이유는 추후에 연구되어야 할 것이다.

Table 2. Apply post-processing (Metric : Dice coef.)

Model	Prediction score	Min_size Filter	Min_size Filter + Convex Hull
VGG19-Unet (1)	0.54904	0.63237	0.63434
MobileNetv2-Unet (2)	0.5507	0.61865	0.62228
EfficientNetB1-Unet (3)	0.58729	0.64324	0.64423
DeeplabV3+ (4)	0.50333	0.5792	0.58869
HRNetV2 (5)	0.59327	0.63061	0.6311

Table 2를 보면 Min_size Filter를 적용했을 때 예측 결과가 크게 좋아졌다. 이를 통해 구름 패턴의 크기는 중요한 특징이며 모델에 의해 예측된 영역이 작을 경우 이를 오탐지로 판단할 수 있음을 알 수 있다. 또한

Convex Hull을 적용했을 때 모든 모델에서 약간씩 성능 향상이 있었다. 즉 윤곽선 개선 및 확장은 모델의 성능에도 좋은 영향을 주고 있었다. 다만 성능이 크게 좋아지지 않은 이유는, 모델이 구름을 오탐지한 경우에서도 Convex Hull을 통해 윤곽선을 확장했기 때문으로 추측된다. 한편 Table 1과 달리 실제 예측에서는 HRNetV2의 결과가 가장 좋았다. 후처리는 예측 이후에 임의적으로 시행하는 작업임을 고려하면, 우리가 사용한 모델 중 자체적인 분할 성능이 가장 좋은 모델은 HRNetV2라고 판단할 수 있다.

Table 3. Ensemble results

Ensemble	Min_size filter	Min_size filter + Convex hull
(3) + (5)	0.64625	0.64699
(2) + (5)	0.63931	0.64027
(3) + (4) + (5)	0.63954	0.64052
(1) + (2) + (3)	0.64905	0.6499
(1) + (3) + (5)	0.64652	0.64725
(2) + (3) + (5)	0.64913	0.64973
(1) + (2) + (3) + (5)	0.64888	0.64954
(1) + (2) + (3) + (4) + (5)	0.64512	0.6458

Table 3에서 앙상블 부분의 숫자는 Table 2에서 정의되었으며, 가장 성능이 좋은 모델인 HRNetV2를 중심으로 조합을 형성하였다. 위 표를 보면 앙상블을 진행했을 때의 성능이 전반적으로 진행 전에 비해 좋아짐을 알 수 있다. 그러나 DeeplabV3+의 성능이 좋지 않았기 때문에 해당 모델을 포함한 조합의 결과는 다른 조합에 비해 좋지 않았다. 예측 결과가 가장 좋은 HRNetV2 없이 세 가지 U-Net 모델의 조합으로 구성된 앙상블에서 가장 좋은 점수를 받은 것이 상당히 특이한 점이지만, 이는 Convex Hull에 의한 결과라는 사실을 감안할 때 U-Net 모델의 조합이 다른 조합에 비해 좋다고는 할 수 없다.

5. 결론

본 논문은 의미론적 분할을 수행하는 딥러닝 모델을 이용해 가시광선 파장으로 제한된 환경에서 위성사진을 통한 구름의 형태 분류를 진행하는 방법을 제시하였다. 우리가 얻어낸 0.6499의 평가 점수는 Kaggle 대회 참가자 1533명의 결과 중 252위에 해당하며 1위에 비해 0.03밖에 차이 나지 않는 좋은 기록이며 딥러닝을 통해 구름 분할을 잘 수행할 수 있음을 보였다. 한편 모델 간 비교에서는 고해상도에서 발견되는 특징을 잃지 않는 HRNetV2의 성능이 가장 뛰어났다. 이를 통해 구름 분할의 핵심은 높은 해상도에서 발견되는 미세한 정보라 판단할 수 있다.

하지만 실제 연구에서는 하드웨어 성능이 좋지 않아 이미지 해상도를 크게 줄일 수 밖에 없어 이런 미세한 정보들을 많이 확보하기 힘들었다는 점이 아쉽다. 최상의 연구 환경이 갖춰졌다면 원본 해상도에서 학습이 가능함은 물론 최신 딥러닝 모델을 실험해볼 수 있었고 초매개변수 튜닝에서의 제약 또한 미미했을 것이다.

또한 우리는 주어진 레이블을 직접 수정해 원본 레이블이 지정한 구름 패턴 영역이 실제보다 크게 설정된 문제를 해결할 수 있는 2가지 아이디어가 있었고 이를 구현하고자 시도했으나 아쉽게도 미완에 그쳤다. 우선 훈련 데이터의 원본 레이블을 이용해 분할 모델을 에폭 1~2 정도로 가볍게 학습시킨 뒤, 훈련용 이미지 내 구름 영역을 해당 모델을 통해 예측한 후 Convex Hull을 진행하는 방법이 있었으나 이 과정으로 수정된 영역이 너무 작아 레이블로 이용할 수 없었다. 두 번째는 각 훈련용 이미지마다 영역 레이블을 1%만 남기고 전부 삭제하여 점들이 흩어져있는 형태로 만든 뒤 Morphological Dilation과 Convex Hull을 통해 해당 점들을 중심으로 영역을 강제로 확장시키는 방법이었고, 이 방법은 실제로 우리가 목표한 수준에 맞는 영역을 새롭게 정의하였으나 아직 이를 통한 모델 학습 결과를 도출하지 못했다. 시간이 더 주어졌다면 충분히 논문에 포함시킬 수 있는 상황이어서 큰 아쉬움이 남는다.

References

Jonathan Long, Evan Shelhamer, and Trevor Darrell. 「Fully Convolutional Networks for Semantic Segmentation」, 2015

성선경 · 최석근 · 최재완. 「전이학습을 활용한 DeeplabV3+ 모델 기반 Plantscope 영상의 구름탐지」, 2020

오한,김유경,김희섭 「다목적 실용위성 3호/3A호 영상에 대한 딥러닝 기반 객체 검출 알고리즘의 성능 비교」, 2021

Yuhang Jiang, Wei Cheng, Feng Gao 등. 「A Cloud Classification Method Based on a Convolutional Neural Network for FY-4A Satellites」, 2022

Andrew G. Howard, Menglong Zhu, Bo Chen 등. 「MotileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications」, 2017

Mingxing Tan, Quoc V.Le. 「EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks」, 2019

Liang-Chieh Chen, Yukun Zhu, George Papandreou 등. 「Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation」, 2018

Stephan Rasp, Hauke Schulz, Sandrine Bony 등. 「Combining crowd-sourcing and deep learning to explore the meso-scale organization of shallow convection」, 2019

Jingdong Wang, Ke Sun, Tianheng Cheng 등. 「Deep High-Resolution Representation Learning for Visual Recognition」, 2019