



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사 학위논문

합성곱 신경망을 이용한 암석 균열
인식 자동화 알고리즘 개발

Automated Rock Fracture Detection
Algorithm with Convolutional Neural
Networks

2019년 8월

서울대학교 대학원
에너지시스템공학부
변 훈

국문초록

암반에 존재하는 균열과 절리는 강도, 탄성계수, 투수계수 등에 큰 영향을 미치기 때문에 이들을 잘 검출하는 것은 중요한 문제이다. 특히 사진측량법은 간단하고 경제성이 있으므로 이를 기반으로 하는 연구들이 이루어졌다. 그 중, 절리는 선형성을 띠기 때문에 비교적 인식이 평이하나, 균열은 비정형성을 보이기 때문에 인식이 상대적으로 어려워 관련 연구가 부족한 실정이다. 또한, 기존의 연구들은 균열 인식을 방해하는 다양한 노이즈가 없는 이미지를 주로 사용하였다. 그림자, 균열 사이의 충전물, 식생 등의 노이즈는 전통적인 알고리즘의 균열 인식 정확도를 낮추는 요인이나, 사진을 촬영한 현장의 환경에 따라 사진에 포함될 수 있다.

본 연구에서는 딥러닝 알고리즘의 일종인 합성곱 신경망을 사용하여 다양한 노이즈가 존재하는 이미지로부터 암석 균열을 자동으로 인식하는 알고리즘을 개발하였다. 딥러닝 기반 합성곱 신경망 알고리즘은 사람이 직접 시행착오를 통해 적절한 피처를 결정했던 전통적인 이미지 처리 방식과 달리 신경망이 스스로 이미지에서 적합한 피처를 추출하여 이용하기 때문에 인식 성능이 향상된다. 또한, 기존 연구들은 모델 개발에 사용한 균열 이미지와 같은 이미지로 테스트를 수행하였기 때문에 그로부터 개발된 모델은 그 특정 이미지에 대해서만 좋은 성능을 낼 수 있었다. 그러나, 본 연구에서는 테스트 과정에 완전히 새로운 균열 이미지를 사용함으로써 새로운 이미지에도 좋은 결과를 내는 것을 보였다. 종합적으로, 본 연구에서 개발된 알고리즘은 사진으로부터 신속하고 일관적으로 균열 인식을 할 수 있으며 다양한 비정형 균열 이미지들에 대해서도 높은 검출 성능을 보인다.

주요어 : 균열, 이미지 인식, 합성곱 신경망

학 번 : 2015-22944

목 차

1. 서론	1
2. 인공신경망 이론	5
2.1 완전연결 신경망	5
2.2 합성곱 신경망	8
3. 실험 방법	13
3.1 데이터 수집	14
3.2 데이터 가공	16
3.2.1 데이터 레이블링	16
3.2.2 데이터 분리	18
3.2.3 데이터 증강	20
3.2.4 데이터 전처리	23
3.3 합성곱 신경망 모델 구조	27
3.4 학습 상세 과정	30
3.5 후처리 과정	30
4. 결과 및 고찰	33
4.1 그림자, 검은 표면 등이 있는 사진	34
4.2 줄무늬 구조가 존재하는 사진	40
4.3 충전물이 존재하는 사진	43
4.4 굵힌 자국이 존재하는 사진	47
4.5 식생이 존재하는 사진	49

4.6 모델 성능 고찰	52
5. 결론	56
참고문헌	58

그림 목 차

Figure 2.1 Architecture of fully connected network	6
Figure 2.2 Schematic representation of convolution operation with 3x3 kernel and bias	9
Figure 2.3 Schematic diagram of max pooling operation with strides 2	11
Figure 3.1 Flowchart of rock fracture segmentation method proposed in this study	14
Figure 3.2 Rock fracture image sampling sites	15
Figure 3.3 Rock fracture images used in this study	16
Figure 3.4 Labeled image	18
Figure 3.5 Split data into training data, validation data and test data	20
Figure 3.6 Procedure of data augmentation.	22
Figure 3.7 Histogram redistribution process	25
Figure 3.8 (a) Raw images obtained on site, and (b) preprocessed images after CLAHE	26
Figure 3.9 Network architecture used in this study	28
Figure 4.1 Rock fracture detection results with black pattern on surface	35
Figure 4.2 Rock fracture detection results with shadow on surface	37
Figure 4.3 Rock fracture detection results with striped pattern ·	41

Figure 4.4 Rock fracture detection results with infill material	45
Figure 4.5 Rock fracture detection results with surface scratch	48
Figure 4.6 Rock fracture detection results with vegetation	51
Figure 4.7 Rock fracture image, its label and prediction results	54

표 목 차

Table 4.1 Rock fracture detection performance of proposed model	55
---	----

1. 서론

암반에 존재하는 균열과 절리는 강도, 탄성계수와 같은 역학적 성질과 투수계수와 같은 수리적 특성에 큰 영향을 끼친다. 이러한 암반의 성질들은 암반 구조물의 안정성 평가, 방사성 폐기물 처분장의 안전성 평가 등에 활용되기 때문에 균열과 절리의 정보를 정확히 파악하는 것은 중요한 문제이다. 이와 같은 필요성을 바탕으로 균열, 절리를 평가하는 다양한 방법들이 연구되어왔다.

특히, 절리는 선형성과 균집성을 띠며, 이러한 특성은 절리 인식 방법에 사용되었다. 절리 조사에는 전통적으로 조사선 조사와 조사창 조사 방법이 사용되어왔다(Wei et al., 1995). 조사선 조사는 노두에 직선의 조사선을 설치한 후 조사선과 교차하는 불연속면들을 분석하는 방법이다. 간단하다는 장점 덕분에 쉽게 적용될 수 있으나, 노두 전체에 대한 정량적인 분석이 아닌 1차원적인 분석만 가능하다는 단점이 있다. 이러한 한계점을 개선한 방법으로 조사창 조사가 있다. 직선을 사용하는 조사선 방식과 다르게 직사각형 형태의 조사창을 설정하며, 조사창 내에 포함된 불연속면 검출을 통해 불연속면 정보를 파악한다. 조사창 조사를 통해 조사선 조사로는 불가능했던 많은 불연속면들을 검출할 수 있으나, 동시에 측정해야 할 불연속면의 개수가 늘어나는 한계점이 있다.

이러한 전통적인 불연속면 조사 방법은 다양한 단점이 존재한다. 사람이 수작업으로 노두 조사를 하며 불연속면을 기록하기에 많은 시간과 비용이 소모되며 노두에의 접근성, 위험성 문제가 존재한다. 또한, 사람이 직접 불연속면 정보를 평가하는 경우, 평가자에 따라 결과에 차이가 나타나며, 이는 암반의 안정성 평가에 개인차를 야기한다. 따라서 불연속면 조사를 신속하고 일관성 있게 하기 위해서는 불연속면 조사와 평가 과정을 자동화할 필요가 있다.

사진측량법은 사람이 접근하기 힘든 곳에 있는 노두에 대해서도 빠르고 정확하게 불연속면 정보를 얻을 수 있는 방법이다(McCarter, 1976).

Deb. et al.(2008)은 사진측량법을 사용하여 노두에서 절리의 디지털 사진을 얻은 후, 캐니 에지 검출(Canny edge detection) 알고리즘을 사용하여 에지 검출을 하고 허프 변환 (Hough Transform)을 사용하여 선형성을 띠는 절리를 추출하였다. Mohebbi et al.(2017)은 노두로부터 절리 사진을 촬영한 후, 전처리 과정을 거친 다음, 마찬가지로 캐니 에지 검출 알고리즘과 허프 변환을 사용하여 절리를 검출하였다. 비록 허프 변환을 사용한 절리 검출은 효과를 보이나, 적용 과정에서 절리의 선형성에 대한 가정이 들어있으므로 균열 검출에는 적용할 수 없는 한계점이 존재한다. 선형성과 균집성을 보이는 절리와 달리 균열은 불규칙하게 존재하며 형태가 정해져 있지 않다. 이러한 특성은 균열 인식을 어렵게 한다.

Wang(2005)은 암반의 균열을 검출하기 위해 먼저 사진을 회색조로 변환한 다음 각 픽셀(pixel)에서 주변 픽셀값과의 차이를 사용하였다. 또한, 이미지의 해상도를 조절해가며 노이즈를 제거했다. 이 방법에는 이미지에서 밝기 차이를 보이는 선은 모두 균열로 인식하는 한계점이 존재하며, 다양한 종류의 암석 사진들에 대해 일반적으로 노이즈 제거에 최적화된 해상도를 제시하지 못했다. Wang et al.(2007)은 서포트 벡터 머신(support vector machine)을 사용하여 균열 인식을 시도하였다. 우선, 캐니 에지 검출을 사용하여 균열과 노이즈를 포함한 에지들을 검출한 다음, 이미지로부터 균열 판별에 필요한 여러 피처(feature)를 추출한 후 서포트 벡터 머신을 사용하여 균열과 노이즈를 분류하였다. 그러나 실험에 사용한 이미지의 수가 적고, 학습용 이미지로 학습시킨 서포트 벡터 머신의 매개변수들을 새로운 암석 이미지에 테스트해보지 않았으므로 성능을 보장할 수 없다. 또한, 서포트 벡터 머신은 균열과 노이즈의 분류에 이용할 피처들의 종류를 직접 결정해야 하므로 저질로 피처들을 추출해주는 딥러닝 기반 방식보다 성능이 떨어진다. He and Wang(2010)은 암석 균열 인식에 인공신경망의 한 종류인 PCNN(pulse coupled neural networks)를 사용하였으며, PCNN 알고리즘은 노이즈를 덜 검출하기 때문에 전통적인 캐니 에지 검출보다 성능이 뛰어나다고 주장했다. 그러나, 특정한 한 장의 사진에 대해서만 실험을 하였기 때문에 신빙성이 떨어지

며, 이미지에서 균열이 두께를 갖는 경우 그 영역을 모두 균열로 인식하지 못하고 암석과 균열의 경계면만 검출하는 단점이 있다. Seetal and Natarajan(2010)은 ARMA(autoressive moving average) 모델을 사용하여 균열을 검출하였다. 그러나 완만한 표면상의 균열 인식에는 성능이 좋지 않았으며, 마찬가지로 단순히 회색조 사진에서 밝기 정보만을 사용하였다. Wang et al.(2012)는 fractional differential operator를 사용하여 기존 에지 검출보다 좋은 성능을 보였다. 그러나 여전히 미세 균열을 잘 검출하는 것에 대한 반사 효과로써 노이즈가 많이 검출되는 문제를 보였다. Liang(2016)은 multiscale enhancement와 ridge detection 알고리즘을 사용하여 균열 구조를 추출하였다. 이 연구에서는 노이즈를 제거하기 위해 역치값을 설정하였으나, 그 값을 새로운 이미지에 대해서 검증해 보는 작업을 수행하지 않았다. 전반적으로 기존의 암석 균열 인식 연구들은 다음과 같은 공통의 한계점이 있다.

- (1) 테스트에 사용한 이미지가 매우 적다.
- (2) 균열 검출을 위해 설정한 알고리즘 내부의 매개변수 값을 새 이미지에 대해 검증하지 않았다.
- (3) 현장에서 취득한 사진에는 균열 인식을 방해하는 다양한 노이즈가 존재할 수 있으나, 노이즈가 거의 없는 사진들로만 테스트하였다.
- (4) 회색조 사진에서 밝기 정보만을 사용하였기에 충전물이 차 있는 균열, 검은 노이즈, 층리와 편리를 실제 균열과 구별하기 어렵다.

본 연구에서는 이러한 한계점을 극복하기 위하여 합성곱 신경망을 사용한 균열 인식 방법을 사용하였다. 최근 이미지 인식 분야에서는 딥러닝 기반의 합성곱 신경망을 사용한 알고리즘이 크게 발전하여 좋은 성능을 보이며, 암석 분야에서도 도입되고 있다. 암석 이미지를 합성곱 신경망으로 분석하는 연구는 화강암 타일의 품질 평가(Ferreira and Giraldi, 2017), 입자 구조 분석(Cheng and Guo, 2017), 공극 분석(Alqahtani et al., 2018), 그리고 탄성과 속도 예측(Karimpouli and Tahmesbi, 2019)

등 다양한 분야에 적용되고 있다. 특히, 본 연구와 같은 이미지 인식 분야에서는 암석 이미지에서 공극을 인식하는 연구(Karimpouli and Tahmesbi, 2019)가 이루어졌다. 그러나 모두에 존재하는 암반의 균열 조사에 최근의 합성곱 신경망 알고리즘을 적용한 사례는 없는 실정이다. 본 연구에서는 암석 균열 인식에 합성곱 신경망을 적용함으로써 암석 균열을 효과적으로 추출할 수 있었다. 이 논문에서 개발한 알고리즘의 특징은 다음과 같다.

- (1) 암석 표면에 검은 무늬, 굵은 자국, 층리와 편리, 균열 사이의 충전물이 존재해도 균열 인식이 가능하다.
- (2) 암석 사진에 식생(vegetation), 그림자가 존재해도 균열 인식이 가능하다.
- (3) 모델 성능 평가 시 학습에 사용하지 않은 새로운 데이터로 테스트함으로써 새로운 이미지에 대한 균열 검출 성능을 입증하였다.

2. 인공신경망 이론

2.1 완전연결 신경망

합성곱 신경망은 인공신경망의 한 종류로서 이미지 처리에 주로 사용된다. 인공신경망은 두 변수 x, y 의 관계를 표현하는 방법의 일종으로서 x 값을 신경망에 입력하면 각 층을 거쳐 출력층에서 얻어진 값이 y 의 예측값이 된다. 인공신경망의 가장 간단하고 기본적인 형태는 완전연결 신경망(fully connected neural network)으로서 입력값과 출력값은 벡터 형태이다(Fig. 2.1). 첫 번째 열은 입력층(input layer)으로서 데이터가 신경망에 입력되는 위치이며 각 노드(node)에 입력 벡터 중 하나의 스칼라값이 입력된다. 마지막 열은 출력층(output layer)으로서 최종 결과를 산출하는 층이다. 입력층과 마찬가지로 각 노드는 하나의 스칼라값을 출력한다. 중간 부분의 열들은 은닉층(hidden layer)으로서 층의 개수와 각 층에서 노드의 개수는 가변적이다. 각 층 사이를 잇는 선분은 이전 층의 노드에서 다음 층의 노드로의 연산식을 의미한다. 세부적으로 각 층은 이전 층의 노드의 값들에 신경망의 매개변수들인 가중치(weight, W)를 곱하고 편향(bias, b)을 더한 연산을 거친 후, 활성화 함수(activation function)를 취한 값을 갖는다. 신경망에서 l 번째 층의 노드들의 값을 벡터로 표현한 것을 x^l 라 하면 각 층에서의 연산은 식 (1)과 같이 이루어진다.

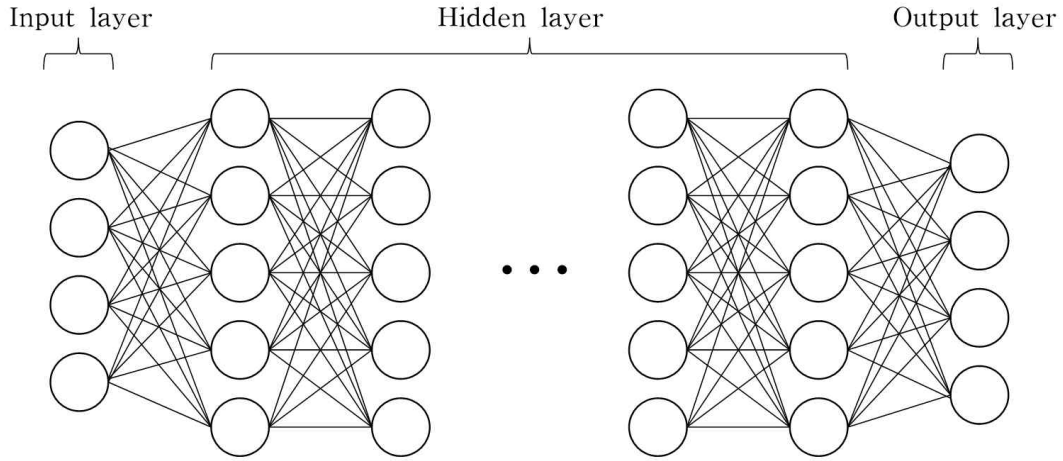


Figure 2.1 Architecture of fully connected network.

$$x^{l+1} = f_{activation}(W^l x^l + b^l) \quad \text{식 (1)}$$

l 번째 층의 노드의 수를 n^l 라 할 경우, W^l 는 (n^{l+1}, n^l) 크기의 행렬이며 b^l 는 길이가 n^{l+1} 인 벡터이다. 활성화 함수로는 시그모이드 함수(sigmoid function), ReLU(Nair and Hinton, 2010) 함수 등이 주로 사용되며 각 함수를 식 (2)에 나타내었다.

$$sigmoid(x) = \frac{1}{1 + e^{-x}} \quad \text{식 (2)}$$

$$ReLU(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}$$

신경망의 학습은 신경망의 매개변수인 가중치와 편향의 값을 결정하

는 과정이다. 신경망의 학습은 지도학습과 비지도학습으로 나눌 수 있으며, 본 연구에서는 지도학습을 사용하였다. 지도학습의 경우 신경망의 마지막 층의 출력값($x^{l_{output}}$)과 실제 정답(y)간의 오차를 최소화시키는 방식으로 학습이 진행된다. 오차를 수치적으로 나타내기 위한 지표로서 손실 함수(loss function, L)을 사용한다. 회귀 문제의 경우 평균 제곱 오차(mean squared error, MSE)가, 분류 문제의 경우 교차 엔트로피 오차(cross entropy error, CEE)가 손실 함수로서 주로 사용된다. 식 (3)은 손실 함수로서 각각 평균 제곱 오차와 교차 엔트로피 오차를 사용하는 경우를 나타낸다.

$W, b = \operatorname{argmin}(L)$, where

$$L(W, b) = \begin{cases} \frac{1}{2} \sum_i^{n_{output}} (y_i - x_i^{l_{output}})^2, & \text{평균 제곱 오차} \\ - \sum_i^{n_{output}} y_i \ln x_i^{l_{output}}, & \text{교차 엔트로피 오차} \end{cases} \quad \text{식 (3)}$$

x_i^l 는 l 번째 층의 i 번째 노드의 값을 의미한다. 신경망 학습은 손실 함수 $L(W, b)$ 를 최소화시키는 것이다. 손실 함수는 아래로 볼록한 함수이므로, $\frac{\partial L}{\partial W} = 0$, $\frac{\partial L}{\partial b} = 0$ 인 지점을 찾으면 손실 함수를 최소로 하는 매개 변수들을 찾을 수 있다. 그러나, 일반적으로 손실 함수는 매우 복잡하고, 매개변수의 공간은 광대하므로 기울기 값 $\frac{\partial L}{\partial W}$, $\frac{\partial L}{\partial b}$ 을 이용하여 손실 함수의 최저위치를 수치적 접근으로 탐색한다. 가장 기본적인 방법으로 사용하는 것이 경사 하강법(gradient descent method)이며 경사 하강법의 경우 매개변수의 업데이트는 식 (4)의 방식으로 이루어진다.

$$W_{new} = W_{previous} - \eta \frac{\partial L}{\partial W_{previous}}, \quad b_{new} = b_{previous} - \eta \frac{\partial L}{\partial b_{previous}} \quad \text{식 (4)}$$

η 는 학습률(learning rate)이며 이 값이 커질수록 매 스텝 매개변수의 업데이트 정도가 커진다. 학습이 진행될수록 매개변수는 손실 함수가 작아지는 값으로 변화하며, 신경망 최적화를 위해서는 주어진 데이터로 여러 번의 반복 학습이 필요하다.

2.2 합성곱 신경망

완전연결 신경망은 벡터 형태의 데이터를 분석하기에 적합하나, 이미지 처리에는 부적절하다. 이미지의 경우 데이터는 픽셀마다 존재하기 때문에 (가로, 세로, 채널)의 3차원 텐서 형태로 존재한다. 이 데이터를 완전연결 신경망에 입력하기 위해서는 1차원 벡터 형태로 변환해야 하며, 이 과정에서 데이터의 형상 정보가 소멸한다. 또한, 픽셀의 수 만큼 입력 층에 노드가 필요하므로, 노드 간의 연결성을 표현하는 매개변수의 수도 매우 많아져 학습이 어려워진다.

이와 같은 완전연결 신경망의 단점으로 인해 이미지 데이터의 경우에는 합성곱 신경망이 주로 사용되며, 본 연구에서도 합성곱 신경망을 이용하였다. 합성곱 신경망에서 각 층 간의 연산은 식 (5)와 같이 이루어진다.

$$x_j^{l+1} = f_{activation} \left(\sum_{i \in C^l} x_i^l * k_{i,j}^l + b_j^l \right), \text{ where } j \in C^{l+1} \quad \text{식 (5)}$$

C^l 는 l 번째 층의 각 채널의 집합을 의미한다. 3차원 텐서 형태인 x^l 는 2차원 텐서들 x_i^l ($i \in C^l$)을 채널 방향으로 쌓은(stack) 것으로 볼 수 있다. 각 x_i^l 는 대응되는 커널(kernel) $k_{i,j}^l$ ($j \in C^{l+1}$)와 합성곱 연산을 한다. 합성곱 연산은 주로 3x3 혹은 1x1 크기의 작은 2차원 텐서인 $k_{i,j}^l$ 가 x_i^l

위를 일정 간격으로 지나가며 각 위치에서 원소별 곱셈(element-wise multiplication)을 한 후 그 총합에 편향을 더한 값을 모든 채널에 대해 수행하고 합하는 것이다. 그 값에 활성화 함수를 취한 후 x_j^{l+1} 의 한 픽셀에 입력한다. 이 연산을 x_i^l 의 모든 픽셀에 대해 수행한다. 합성곱 연산을 통해 얻은 x_j^{l+1} 을 $l+1$ 번째 층에서의 j 번째 피쳐맵(feature map)이라 한다. 각 층에서의 피쳐맵의 수는 신경망 구조를 통해 조절할 수 있다. 합성곱 연산을 통해 이미지의 피쳐는 피쳐맵 형태로 얻어지며, 피쳐맵이 이미지 분석에 활용된다. 합성곱 연산을 도식화한 그림을 Fig. 2.2에 나타내었다.

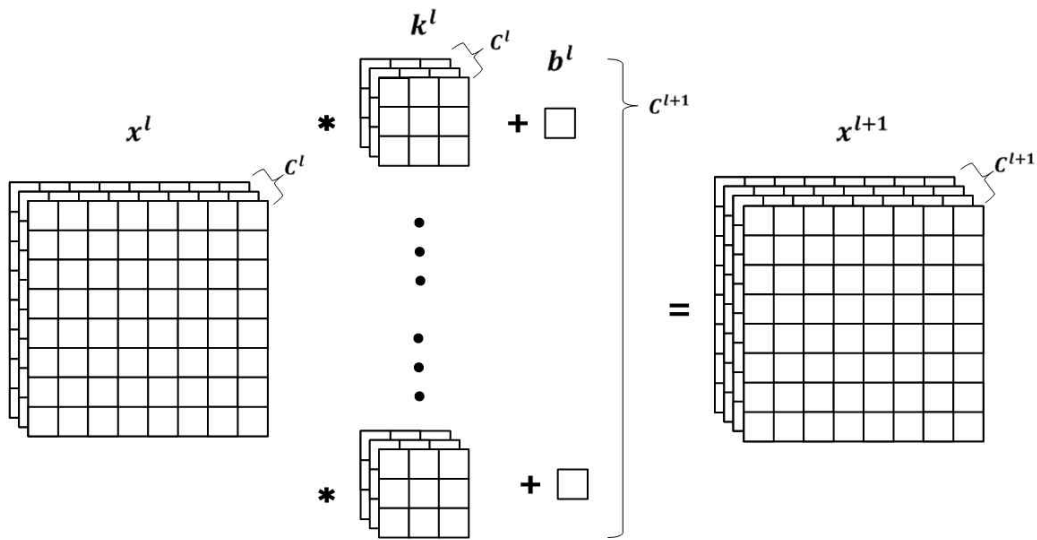


Figure 2.2 Schematic representation of convolution operation with 3x3 kernel and bias.

입력 이미지와 커널의 합성곱 연산을 통하여 피처를 추출하는 합성곱 신경망의 특징상, 추출할 수 있는 피처의 최대 크기는 커널의 크기와 동일하다. 암석의 균열 여부를 판별하기 위해 이미지에서 큰 영역에 대한 피처 추출이 필요하나, 일반적으로 사용되는 3x3 사이즈의 커널을 사용하는 경우 매우 지역적인 피처만 추출할 수 있다. 이처럼 피처 추출을 위한 영역인 수용 영역(receptive field)을 넓히는 것은 중요한 문제이다. 그러나 수용 영역을 높이기 위한 목적으로 단순히 커널의 사이즈를 키우면 동시에 학습해야 할 매개변수의 양이 늘어나게 되며, 신경망의 학습이 어려워진다. 따라서 수용 영역을 넓히기 위한 방법으로 풀링(pooling)을 사용한다.

풀링은 이미지의 가로, 세로 방향 사이즈를 줄이는 연산이다. 풀링을 통해 원본 이미지보다 사이즈가 줄어들었기 때문에 풀링 연산 후에 합성곱 연산을 하는 경우 원본 이미지 대비 수용 영역이 그만큼 늘어나게 된다. 따라서 일반적으로 합성곱 연산과 풀링을 여러 번 반복하는 형태로 신경망을 구성한다. 이 경우 초반부의 합성곱 연산에서는 작은 사이즈의 피처를 추출하고, 후반부의 합성곱 연산에서는 큰 사이즈의 피처를 추출할 수 있다. 풀링 연산으로는 보통 최대 풀링(max pooling) 방식이 사용되며(Scherer et al., 2010), 식 (6)과 같이 표현된다.

$$x_{i,j}^{l+1} = \max_{p \in [n(i-1)+1, nj+1], q \in [n(j-1)+1, nj+1]} (x_{p,q}^l) \quad \text{식 (6)}$$

x^l 는 풀링 연산 전의 입력 이미지이며, x^{l+1} 는 풀링 연산 후의 출력 이미지이다. n 은 스트라이드(stride)를 나타낸다. 풀링 연산은 입력 이미지의 $n \times n$ 영역의 픽셀들의 값 중 최대값을 출력 이미지의 픽셀값으로 사용하며 출력 이미지의 가로, 세로 길이를 각각 n 배씩 축소한다. 풀링 연산을 그림으로 나타내면 Fig. 2.3과 같다.

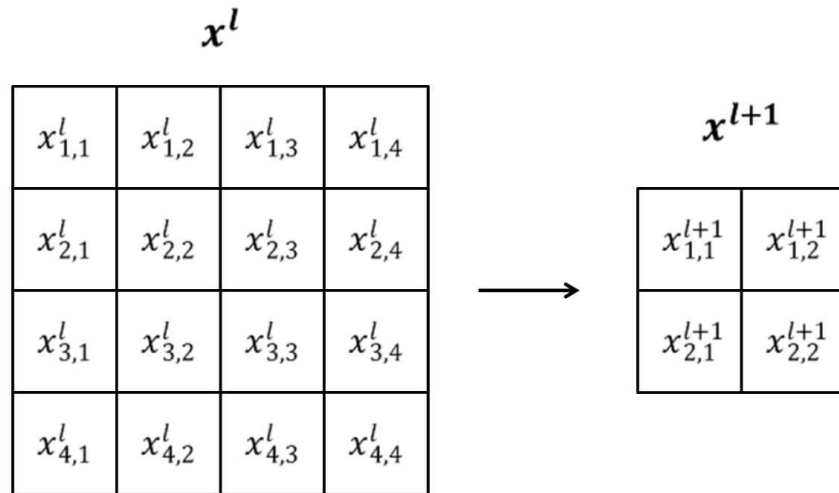


Figure 2.3 Schematic diagram of max pooling operation with strides 2. Both width and height decrease in half.

이미지 처리에서는 이미지의 피처를 잘 추출하는 것이 중요하다. 전통적인 이미지 처리 방식에서는 목적에 맞게 추출할 이미지의 피처를 사람이 직접 정하였으며, 대표적으로 SIFT(Scale-Invariant Feature Transform), HOG(Histogram of Oriented Gradient) 등의 피처가 사용되었다 (Lowe, 1999, Dalal and Triggs, 2005). 절리를 인식하고자 할 때, 암석의 기질(matrix)과 구분하기 위해서는 선형성 특징이 효과적이거나, 편리와 구분할 때는 선형성이 아닌 다른 특징 사용하여야 한다. 즉, 전통적인 알고리즘 하에서는 이미지로부터 어떤 특징을 사용할지도 중요한 고려사항이었다. 일례로, Wang(2007)은 균열 검출을 위해 서포트 벡터 머신에 사용하기 위한 피처를 직접 결정하고 추출하였다.

합성곱 신경망을 사용하는 경우 커널과 합성곱 연산을 수행하여 얻은 피처맵이 피처가 된다. 새로운 커널과 연산을 한 경우에는 새 피처가 추출된다. 신경망 학습은 커널의 원소값을 결정하는 과정이므로, 학습이 진

행될수록 목적에 맞는 커널과 그에 의한 피처를 얻을 수 있다. 즉, 기존에는 사람이 어떤 피처를 사용할지를 시행착오를 통해 정하였다면 합성곱 신경망을 사용하는 경우에는 신경망이 학습하며 스스로 적절한 피처를 찾아가는 종단간 학습(end-to-end learning)이 가능해졌다. 결국, 전통적인 알고리즘보다 합성곱 신경망을 사용한 경우 목적에 맞는 피처를 사용할 수 있으며 이는 성능 향상으로 이어진다.

3. 실험 방법

적은 수의 테스트 이미지로 인한 신뢰도 하락, 학습 데이터와 테스트 데이터의 미분리, 균열과 노이즈의 구분 성능 저하 등은 기존 알고리즘들의 공통적인 문제이다. 본 연구에서는 다량의 암석 균열 데이터를 취득하여 사용하였다. 또한, 단순 회색조 이미지가 아닌 *RGB* 이미지를 사용함으로써 더욱 고차원적인 균열 판정이 가능하도록 하였다. 그리고 합성곱 신경망을 사용함으로써 일일이 피처를 결정할 필요 없는 종단간 학습을 가능하게 하여, 균열 인식 과정에서 사람의 개입을 줄였으며 검출 성능을 향상시켰다. 또한, 학습 데이터와 테스트 데이터를 분리하여 테스트 결과의 신뢰성을 높였다. 본 연구의 전체 과정을 Fig. 3.1에 나타내었다.

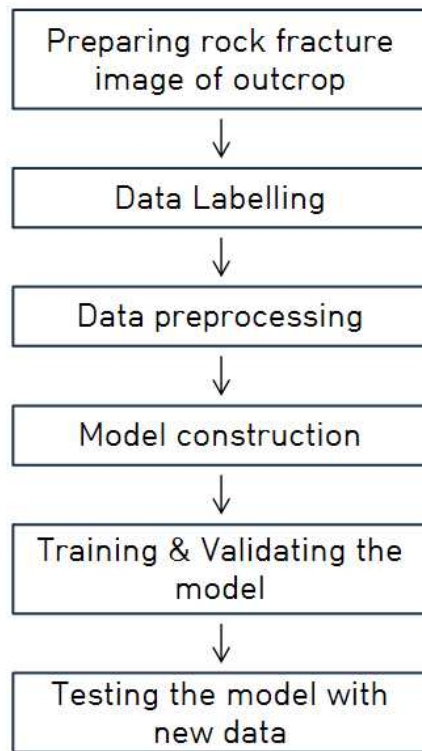


Figure 3.1 Flowchart of rock fracture segmentation method proposed in this study.

3.1 데이터 수집

본 연구에서는 직접 국내의 다양한 지역에서 노두 조사를 통해 암반 균열 데이터를 수집하였다. 모델 개발에 사용된 암반은 북한산, 관악산 그리고 정선군의 남면, 화암면 일대에 분포한다(Fig. 3.2). 야외 조사를 통해 각 지역의 노두들을 관찰한 후 암반 균열 사진들을 촬영하였다.

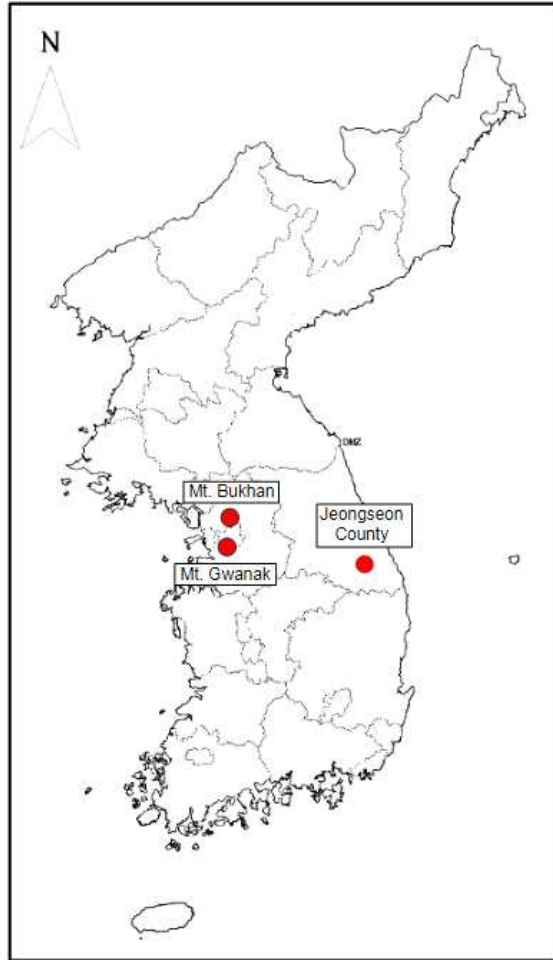


Figure 3.2 Rock fracture image sampling sites.

각 사진에는 다양한 크기, 빈도, 방향의 균열과 절리뿐 아니라 충전물, 흙, 수목, 그림자도 일부 포함되었다. 이처럼 노두에서 관찰되는 암반의 상태를 가공하지 않고 직접 모델 학습에 사용하였으며, 최종적으로 개발할 모델도 이와 같은 노이즈가 존재하는 사진으로부터 균열을 인식할 수 있도록 하였다. Fig. 3.3은 현장에서 채취한 암석 균열 사진들이며 다양한 노이즈를 포함한다.



Figure 3.3 Rock fracture images used in this study.

3.2 데이터 가공

3.2.1 데이터 레이블링

신경망 학습을 위해서는 암석 균열 이미지 외에 균열 부분을 추출한 데이터가 필요하다. 이를 레이블(label) 데이터라고 하며, 레이블 데이터를 만드는 과정을 레이블링(labeling)이라 한다. 원본 사진에서 균열 부분을 수작업으로 파악하여 레이블링하면 레이블 데이터를 만들 수 있으며, 신경망은 균열 이미지와 레이블을 사용하여 학습된다. 따라서 레이블링 과정에서 균열을 결정하는 정확도가 신경망의 성능에 큰 영향을 미친다. 원본 사진에는 균열뿐 아니라 균열 인식을 어렵게 하는 다양한 노이즈들이 존재하는데 레이블링 과정에서 이들을 균열과 잘 구분하는 것이 중요하다.

먼저 암석 균열 이미지의 크기를 1100x1100으로 낮춘 후에 레이블링을 진행하였다. 레이블링은 오픈 소스 파이썬 라이브러리(Wada, 2016)를 사용하여 수행하였다(Fig. 3.4). 이 과정은 암반의 균열 영역을 다각형을 사용하여 나타내는 과정으로서 균열을 더욱 정교하게 검출하기 위해서는 다각형의 각의 개수를 늘리며 균열과 암석의 기질을 세밀하게 분리해야 한다.



labelling fractures on the rock



Figure 3.4 Labeled image. Rock fractures were segmented by tiny polygons.

3.2.2 데이터 분리

신경망 학습을 진행하면 신경망은 학습에 사용한 데이터에 대해서 좋

은 성능을 내나, 새로운 입력 데이터를 사용할 때 성능이 현저히 떨어지는 과적합(overfitting) 현상을 보이곤 한다. 이는 신경망이 오직 학습에 사용된 데이터만 잘 예측하도록 학습되었고, 신경망 구조를 결정할 때 사용되는 매개변수 값들을 선택하는 과정에서 학습용 데이터에 유리하도록 편향된 선택을 하기 때문이다. 이러한 현상을 방지하기 위해 데이터를 학습-검증 데이터와 테스트 데이터로 분리하였다. 신경망 학습에는 학습-검증 데이터 68장을 사용하였고, 학습 전 과정에서 테스트 데이터의 어떠한 정보도 사용하지 않았다. 신경망 개발 과정에는 학습-검증 데이터만 사용하며, 학습 데이터와 검증 데이터를 3:1의 비율로 나누어서 사용하였다. 신경망의 구조를 구성하는 하이퍼 파라미터를 설정할 때는 학습 데이터로 학습을 시킨 후 검증 데이터로 테스트를 한 결과를 기반으로 여러 후보 구조 중 최종 신경망 구조를 선택하였다. 그러나 이 과정에서 특정 검증 데이터에 유리하도록 신경망 구조를 결정하는 문제가 발생할 수 있다. 이를 방지하기 위해 학습 과정에 4중 교차 검증(4-fold cross validation)을 적용하였다. 즉, 학습-검증 데이터를 네 종류의 3:1 비율로 나눈 후 학습-검증 과정을 거친 다음 네 경우에 평균적으로 높은 성능을 보이는 신경망 모델을 최종적으로 선택하였다. 학습-검증 데이터로 신경망의 구조를 확정된 후에는 새로운 테스트 데이터를 사용하여 모델의 성능을 평가했다. 이러한 데이터 분리 과정을 Fig. 3.5에 나타내었다. 결과적으로 본 연구에서 개발한 신경망 모델은 특정 데이터에 대해서만 좋은 성능을 보이지 않으며, 새로운 암석 사진의 균열 인식에도 적용될 수 있다.

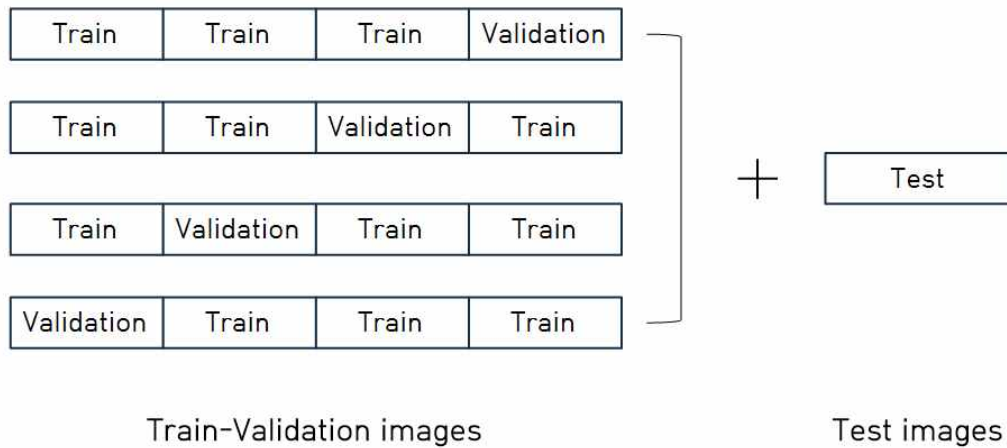


Figure 3.5 Split data into training data, validation data and test data. 4-fold cross validation was conducted.

3.2.3 데이터 증강

신경망 학습에 사용하는 데이터의 양이 많을수록 신경망의 성능은 향상된다. 그러나, 암석 균열 데이터 수집의 물리적인 한계로 인하여 학습에 사용할 수 있는 데이터의 양은 제한적이다. 학습에 사용하는 암석 균열 데이터가 부족한 경우에는 신경망이 균열의 일반적인 특징을 올바르게 학습하지 못하는 문제가 발생한다. 이러한 문제를 해결하는 가장 쉽고 일반적인 방법은 인공적으로 데이터에 레이블-보존(label-preserving) 변환을 수행하는 데이터 증강(data augmentation) 작업을 시행하는 것이다(Simard et al., 2003).

본 연구에서는 데이터 증강으로서 flip과 random crop을 사용하였다. 먼저 학습-검증용 암석 균열 이미지 1100x1100 사이즈 68장의 가장자리 부분을 제거하여 1024x1024 사이즈로 변환하였다. 가장자리 부분을 제거

하는 이유는 레이블링 과정에서 현실적으로 사진의 가장자리 부분에 있는 균열을 확실히 구별하고 레이블링하는 것이 불가능하기 때문이다. 그 후, 이미지를 네 방향으로 flip하여 이미지의 수를 4배로 증가시켰다. 그리고 각각의 flip된 이미지들을 가로, 세로 4등분씩 하여 256x256 사이즈로 변환하며 이미지의 수를 다시 16배 증가시켰다. 마지막으로 256x256 사이즈의 각 이미지에서 224x224 사이즈로 random crop을 수행하여 이미지 수를 2배 더 증가시켰다. 종합하면, 학습-검증용의 1024x1024 사이즈 이미지 68장은 224x224 사이즈 이미지 8704장으로 128배 증가하였다 (Fig. 3.6).

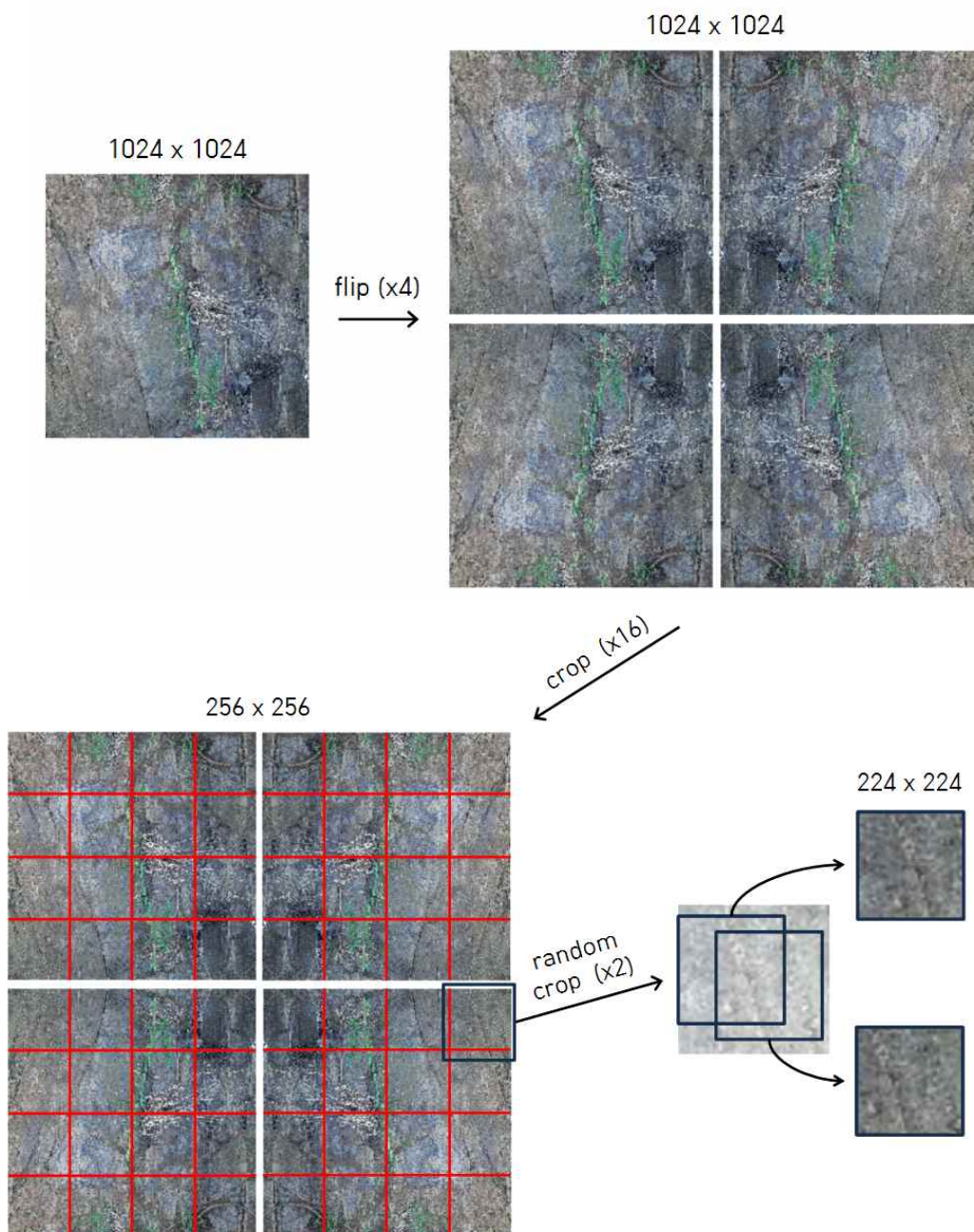


Figure 3.6 Procedure of data augmentation.

3.2.4 데이터 전처리

야외 현장에서 사진 촬영을 할 때의 기상 상황과 조명에 따라 균열 인식 난이도는 크게 좌우될 수 있다. 적절한 조명하에서 촬영하였다면 균열의 특징이 도드라지나, 조명이 적절치 못한 경우에는 특정 영역 또는 전체가 매우 밝거나 어두운 사진이 얻어지며 이 경우 균열 검출 성능이 제한된다. 이와 같은 문제를 해결하기 위한 방안으로 대비제한 적응 히스토그램 평활화(contrast limited adaptive histogram equalization, CLAHE)를 적용하였다(Pizer et al., 1987). 대비제한 적응 히스토그램 평활화는 이미지에서 픽셀값들의 분포를 평탄화함으로써 이미지의 전반적인 대비(contrast)를 향상시켜 이미지 처리에 유리하도록 변환하는 작업이다.

대비제한 적응 히스토그램 평활화를 적용하기 위해 먼저 *RGB* 형태의 이미지를 *HSV* 색 공간으로 변환시켰다. *HSV*의 구성요소인 *H*는 색상(Hue)을, *S*는 채도(Saturation)를, *V*는 명도(Value)를 나타낸다. *RGB* 이미지를 *HSV*로 변환하는 과정은 식 (7)과 같다(Nishad, 2013).

$$\begin{aligned}
& V = \max(R, G, B) \\
& S = \begin{cases} \frac{V - \min(R, G, B)}{V} \times 255 & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases} \\
& \text{if } S = 0; \\
& \quad H = 0 \\
& \text{else} \\
& \quad \text{if } V = R; \\
& \quad \quad H = \frac{G - B}{R - \min(R, G, B)} \\
& \quad \text{elseif } V = G; \\
& \quad \quad H = 2 + \frac{B - R}{G - \min(R, G, B)} \\
& \quad \text{else;} \\
& \quad \quad H = 4 + \frac{R - G}{B - \min(R, G, B)} \\
& \text{end} \\
& \text{if } H < 0; \\
& \quad H = H + 6 \\
& \text{end} \\
& H = H^* \frac{255}{6} \\
& \text{end}
\end{aligned}
\tag{7}$$

이후, 대비제한 적용 히스토그램 평활화는 다음의 과정을 통해 이루어진다. 먼저, 히스토그램 평활화는 일정 크기의 타일(tile) 단위로 이루어지기 때문에 이미지를 20x20 사이즈의 영역으로 나눈다. 그 후, 각 타일에서 V 값의 히스토그램을 구한다. 다음으로 히스토그램에서 특정 값(Clip Limit) 이상의 분포를 전 영역에 고르게 분산시키는 과정을 수행한다(Fig. 3.7). 이 과정은 향후 계산할 누적 분포 함수(cumulative distribution function, CDF)의 기울기를 낮춤으로써 극단적으로 밝거나 어두운 노이즈가 강조되지 않도록 한다.

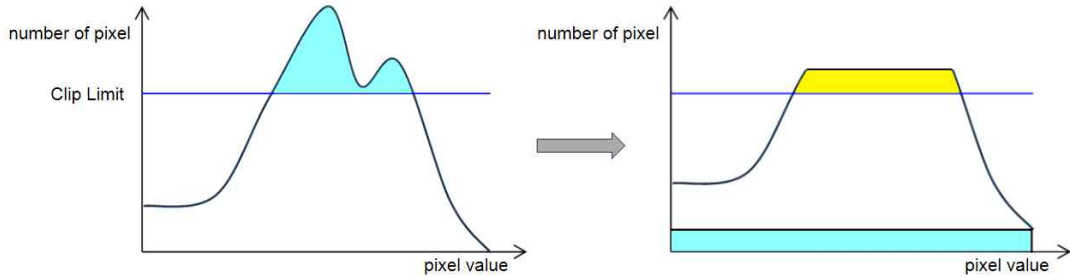


Figure 3.7 Histogram redistribution process.

이후 수정된 히스토그램의 누적 분포 함수를 식 (8)과 같이 계산한다.

$$cdf(x) = \sum_{i=1}^x n_i \quad \text{식 (8)}$$

n_i 는 타일에서 V 값이 i 인 픽셀의 개수이다. 히스토그램 평활화 작업은 식 (9)와 같이 수행된다.

$$V(x) = \frac{cdf(x) - \min(cdf)}{\max(cdf) - \min(cdf)} \times 255 \quad \text{식 (9)}$$

$V(x)$ 는 평활화 작업 전, V 값이 x 였던 픽셀의 새로 계산된 V 값이다. 마지막으로 양선형 보간법(bilinear interpolation)을 사용하여 각 타일간의 경계를 매끄럽게 한다.

부적절한 조명 조건에서 찍힌 사진도 대비제한 적응 히스토그램 평활화를 사용함으로써 선명도를 높일 수 있으며, 이 과정을 통해 물체 인식 성능이 향상된다. 원본 이미지와 대비제한 적응 히스토그램 평활화 작업을 수행한 이미지들을 비교한 그림을 Fig. 3.8에 나타내었다.



(a)

(b)

Figure 3.8 (a) Raw images obtained on site, and (b) preprocessed images after CLAHE.

3.3 합성곱 신경망 모델 구조

개발한 신경망의 구조는 전형적인 인코더-디코더(encoder-decoder) 형태인 U-Net을 기본으로 한다(Ronneberger et al., 2015). U-Net은 생의학 이미지에서 세포벽 검출을 하는 목적으로 만들어진 모델로서 이진 분류(binary classification)에 적용하기 위해 고안되었다. 암석에서 균열을 검출하는 과정도 이진 분류 문제이므로 중추 구조(backbone network)로서 U-Net을 사용하였다. U-Net과 같은 인코더-디코더 형태의 신경망은 다운샘플링(downsampling)이 이루어지는 인코더와 업샘플링(upsampling)이 이루어지는 디코더 부분이 연결된 형태이다. 신경망의 전반부인 인코더는 입력 이미지에서 고차원의 피쳐 정보를 얻는 역할을 하는 부분으로써 풀링을 통해 이미지를 다운샘플링하며 연속적인 합성곱 연산으로 피쳐를 추출한다. 그리고 축소된 이미지는 신경망의 후반부인 디코더에서 업샘플링되며 입력 이미지 크기로 회복된다. 이처럼 합성곱 연산을 연속적으로 수행하면 더욱 고차원의 피쳐를 얻을 수 있기 때문에 복잡한 형태의 이미지에서도 균열과 배경을 구분할 수 있다.

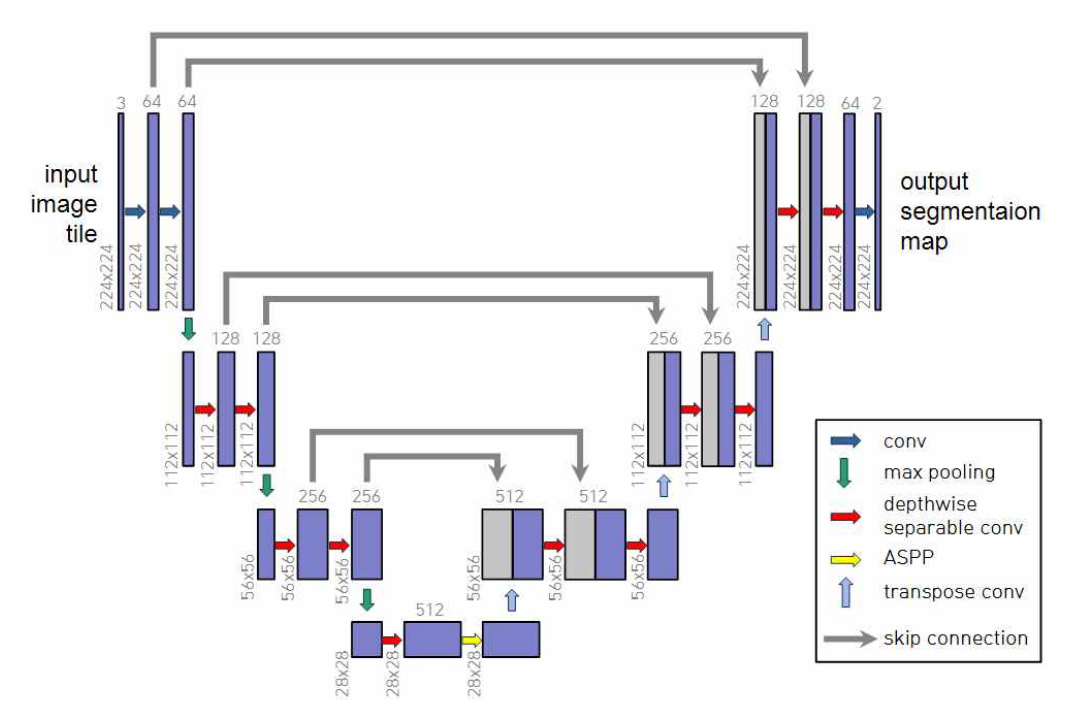


Figure 3.9 Network architecture used in this study. Every convolution layer consists of 3x3 convolution except the final convolution layer which yields the output segmentation map. Each box denotes the feature maps except the initial and final ones. The number located on the topside of the box stands for the number of feature maps.

본 연구에서 사용한 신경망은 U-Net 모델을 수정하여 개선하였다 (Fig. 3.9). 우선, 다운샘플링 과정에서 생성된 피처를 업샘플링 과정에 직접 적용하기 위해 사용하는 스킵 연결(skip connection)의 횟수를 늘렸다. 이로 인해 신경망 학습 단계에서 오차 정보가 신경망 초반부에 있는 매개변수들에 직접 전달되게 함으로써 학습 효율이 증대되어 신경망의 성능이 향상되었다. 또한, 일부 합성곱 연산을 depthwise separable convolution으로 변경하여 매개변수의 수를 크게 줄였다(Howard et al., 2017). 일반적인 합성곱 연산은 공간 방향으로 원소별 곱셈과 채널 방향

으로 합을 동시에 수행하나, depthwise separable convolution은 두 연산을 depthwise convolution과 separable convolution으로 나누어서 수행한다. 그 결과 일반적인 합성곱 연산과 비교하여 매개변수의 개수 대비 더욱 좋은 성능을 낼 수 있다. 다음으로, 기존 U-Net의 최하단부의 다운샘플링-업샘플링 부분을 ASPP(Atrous Spatial Pyramid Pooling)로 대체하였다(Chen et al., 2017). U-Net 구조의 특징상 피쳐맵의 사이즈를 반으로 줄인 후 다시 높이는 과정을 반복하게 되는데, 이 과정에서 이미지의 해상도가 떨어지는 문제가 발생한다. ASPP는 atrous convolution이 병렬로 이루어지는 구조이다. Atrous convolution은 기존 합성곱 연산과 달리 합성곱 연산 시 커널의 크기를 확대시킨 상태로 연산을 수행한다. 그 결과, 풀링을 통한 다운샘플링 없이도 수용 영역을 높일 수 있어, 이미지의 해상도를 감소시키지 않으며 정교한 검출이 가능하게 한다. 마지막으로 모든 합성곱 연산 후에 배치 정규화 작업을 수행하였다(Ioffe and Szegedy, 2015). 합성곱 연산과 활성화 함수를 거치게 되면 각 픽셀의 값들이 고르게 분포하지 않는 현상이 발생하는데 배치 정규화는 이를 예방해준다. 이와 같은 모델 개선 작업을 통해 새로 개발된 신경망은 단순 U-Net을 사용했을 때 보다 암석 균열 인식 성능이 더욱 향상되었다.

Fig. 3.9에서 하나의 박스는 입력 이미지와 피쳐맵을 나타낸다. 신경망이 이미지로부터 균열 인식을 하는 과정은 다음과 같다. 먼저, 신경망에 $224 \times 224 \times 3$ (RGB 3채널) 형태의 이미지가 입력된다. 이후 두 번의 합성곱 연산과 한 번의 최대 풀링이 이루어진다. 이를 통해 이미지의 크기는 반으로 작아지는 다운샘플링이 이루어지며, 피쳐맵의 수는 64개로 증가한다. 그 후, 두 번의 depthwise separable convolution과 한 번의 최대 풀링이 두 번 반복된다. 각 과정에서 피쳐맵의 수는 두 배 늘어나며, 피쳐맵의 사이즈는 반으로 줄어든다. 그리고 한 번의 depthwise separable convolution과 ASPP를 수행하며 고차원의 피쳐맵을 추출하는 과정을 진행한다. 이어서, 업샘플링이 진행된다. 두 번의 depthwise separable convolution과 한 번의 역합성곱(transpose convolution)을 통해, 피쳐맵의 개수는 반으로 줄어들고 크기는 두 배로 늘어나는 작업이 총 세 번

반복된다. Depthwise separable convolution을 수행하기 직전에, 입력 피쳐맵에 대응하는 다운샘플링 과정에서의 동일한 사이즈의 피쳐맵이 스킵 연결을 통해 더해(concatenation)진다. 마지막으로 한 번의 합성곱 연산을 통해 최종 64개의 피쳐맵을 사용하여 픽셀 단위로 암석의 균열 여부를 출력한다. 모든 합성곱 연산에서는 3x3 커널을 사용하나, 마지막 합성곱 연산에서만 1x1 커널을 사용한다. 또한, 모든 합성곱 연산 후에 배치 정규화 작업과 ReLU 활성화 함수를 적용하였다.

3.4 학습 상세 과정

전처리를 마친 데이터를 사용하여 신경망 학습을 진행하였다. 손실 함수로는 교차 엔트로피 오차를 변형한 가중치 교차 엔트로피(weighted cross entropy) 오차를 사용하였다. 이미지에서 대부분 영역은 균열이 아니므로 단순히 교차 엔트로피 오차를 사용하여 학습하는 경우 클래스 불균형 문제(class imbalance problem)에 의해 대부분 영역이 균열이 아닌 것으로 예측되도록 신경망이 잘못 학습된다. 따라서 각 픽셀의 오차값을 계산하는 과정에서 정답 레이블링(ground truth) 값이 균열이면 0.7, 균열이 아닌 경우 0.3의 가중치를 주어 균열 부분의 예측이 더욱 잘 이루어지도록 하였다. 신경망의 매개변수 업데이트 방식은 아담 최적화(Adam Optimizer)를 사용하였다(Kingma and Ba, 2014). 학습 과정은 파이썬(Python)의 텐서플로(Tensorflow) 라이브러리를 사용하여 이루어졌다. 성능 평가의 지표로는 IoU(Intersection Over Union), Precision, Recall을 사용하였다.

3.5 후처리 과정

이미지가 신경망을 통과할 때 이미지의 크기는 1/8로 축소되었다가 다시 확장된다. 비록 스킵 연결과 ASPP를 적용하였으나 여전히 이미지의 픽셀 단위의 해상도가 떨어지는 문제가 발생한다. Chen et al.(2018)은 신경망의 출력값(output segmentation map)에 후처리로서 fully connected CRF(conditional random field) (Krähenbühl and Koltun, 2011)를 접목함으로써 성능 향상을 이루어냈으며, 본 연구에서도 이를 적용하였다. Fully connected CRF는 식 (10)의 에너지 함수를 사용한다.

$$E(x) = \sum_i \phi_u(x_i) + \sum_{i,j} \phi_p(x_i, x_j) \quad \text{식 (10)}$$

i, j 는 이미지의 각 픽셀을 의미하며 x_i 는 픽셀 i 의 최종적으로 예측된 레이블이다. $\phi_u(x_i)$ 는 unary potential로서 $\phi_u(x_i) = -\log p(x_i)$ 를 만족한다. $p(x_i)$ 는 신경망이 픽셀 i 의 레이블을 x_i 로 예측하는 확률이다. $\phi_p(x_i, x_j)$ 는 pairwise potential로서 식 (11)을 만족한다.

$$\begin{aligned} \phi_p(x_i, x_j) = \mu(x_i, x_j) & \left[w_1 \exp\left(-\frac{|p_i - p_j|^2}{2\sigma_a^2} - \frac{|I_i - I_j|^2}{2\sigma_b^2}\right) \right. \\ & \left. + w_2 \exp\left(-\frac{|p_i - p_j|^2}{2\sigma_c^2}\right) \right] \end{aligned} \quad \text{식 (11)}$$

$\mu(x_i, x_j)$ 는 $x_i = x_j$ 일 때 0이고, $x_i \neq x_j$ 일 때 1의 값을 갖는다. p_i, p_j 는 각각 x_i, x_j 픽셀의 좌표값을, I_i, I_j 는 RGB 값을 의미한다. 하이퍼 파라미터인 $w_1, w_2, \sigma_a, \sigma_b, \sigma_c$ 의 값에 따라 신경망의 예측 결과, 픽셀의 좌표, 그리고 픽셀의 RGB 값이 각 픽셀의 균열 여부 최종 판단에 각각 영향을 미치는 정도가 결정된다. Fully connected CRF의 적용 과정은 mean field approximation(Krähenbühl and Koltun, 2011) 알고리즘을 통해 이루어졌다. Fully connected CRF의 하이퍼 파라미터들도 검증 데이터를

통해 결정하였다. 최종 테스트에는 $w_1 = 10$, $w_2 = 0.35$, $\sigma_a = 1.6$, $\sigma_b = 300$, $\sigma_c = 0.35$ 를 사용하였다.

4. 결과 및 고찰

다양한 암석 균열 이미지로 전처리, 합성곱 신경망 연산, 후처리 과정을 거친 후 암석 균열 인식 테스트를 진행하였다. 테스트 과정에는 신경망 학습과 CRF 파라미터 결정에 사용되지 않은 새로운 이미지를 사용하였다. 정선군, 북한산, 관악산에서 수집한 18개의 사진을 테스트에 사용하였다. 또한, 다양한 노이즈가 존재하는 암석 균열 사진으로 모델 성능 평가를 하기 위한 목적으로 화성시 부근에서 노이즈가 포함된 5장의 사진을 추가로 수집하여 총 23장의 사진을 테스트에 사용하였다. 각 사진에는 균열 인식을 어렵게 하는 다양한 상태 조건들이 존재한다.

암석 사진을 통한 관찰로는 균열 판정이 모호한 경우가 일부 존재한다. 이러한 문제가 발생하는 이유는 다음과 같다. 첫째, 균열이 성장하여 원래의 암석이 서로 다른 두 암석으로 분리되는 경우 더 이상 균열이라 보기 힘들다. 둘째, 암석 표면이 굽힌 자국이 깊어지는 경우 균열과 구별하기 힘들다. 이와 같은 경계 문제들은 정확한 균열 검출과 성능 테스트를 방해한다. 이를 방지하기 위해 본 연구에서는 균열의 기준을 다음과 같이 설정하였다.

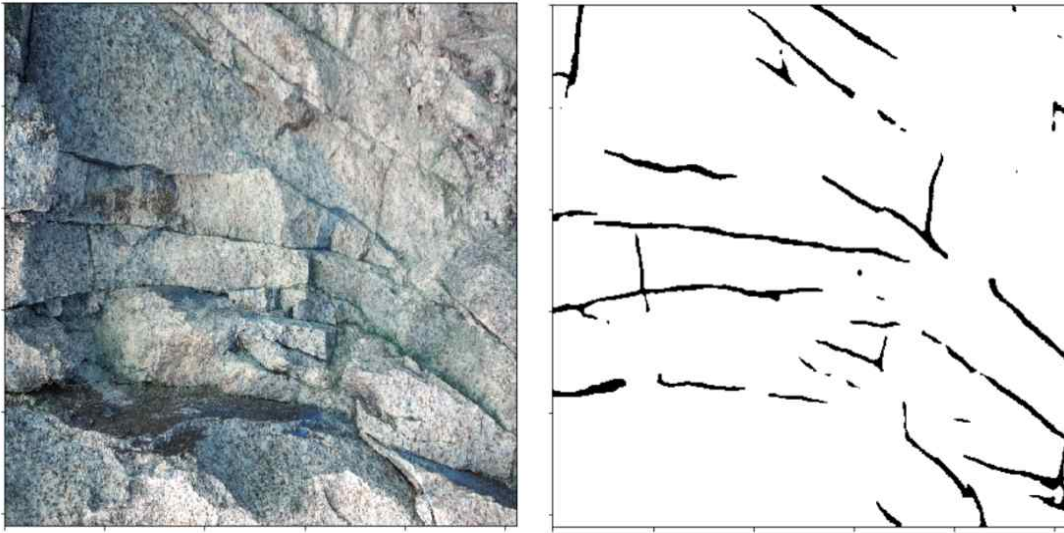
- (1) 균열이 완전히 발달하여 하나의 암석이 두 개 이상의 암석으로 나뉘기 전 단계의 균열
- (2) 단순 표면의 굽힌 자국이 아닌, 깊이감이 있는 균열
- (3) 입체감이 있는 단면과 구분되는 선 구조의 균열

위의 기준을 만족하는 균열을 정확히 인식하기 위해 모델을 학습할 때에 사용한 데이터도 동일한 기준으로 레이블링하였다. 모호한 상태의 균열과 다양한 노이즈가 존재하는 이미지로 테스트를 수행함으로써 제안된 모델의 균열 인식 성능을 분석하였다.

4.1 그림자, 검은 표면 등이 있는 사진

우선, 암석 균열과 비슷하게 검은색을 띠기 때문에 균열 인식을 방해할 수 있는 그림자, 암석 표면의 검은 무늬들이 존재하는 경우에 대해 테스트를 진행하였다. 테스트를 수행한 사진은 총 8장이며, 그 결과를 Fig. 4.1, Fig. 4.2에 나타내었다.

Fig. 4.1은 암석 표면에 검은 무늬가 존재하는 경우이다. 표면의 검은 부분은 균열이 아닌 기질이다. 그러나, 근처 픽셀값을 사용하여 픽셀값 변화로부터 균열을 판정하는 과거 알고리즘들을 사용하는 경우, 균열로 인식될 수 있다. (a)는 좌측 하단부에 검은 기질이 존재한다. 픽셀값의 변화량을 바탕으로 균열을 검출할 경우 이 검은 기질의 모든 둘레는 균열로 인식된다. 그러나, 이 기질을 입체적인 관점에서 보면, 이미지상에서 기질의 하단부는 암석의 모서리일 뿐 균열이 아니다. 따라서 검출 결과도 균열이 아닌 것으로 나타났다.



(a)



(b)

Figure 4.1 Rock fracture detection results with black pattern on surface. Raw images of rock fractures used in the test and corresponding results of fracture segmentation.

Fig. 4.2는 암석 사진 촬영 도중 발생한 그림자가 존재하는 경우이다. 노두에서 암석 사진 촬영을 할 경우, 특수 목적용 조명 장비를 사용하지 않는다면 야외 조명의 방향성에 의해 발생할 수 있는 문제이다. (a), (b)는 암석 표면에 나뭇잎에 의한 그림자가 존재하는 경우이다. 또한, (c)~(f)는 암석 자체의 요철에 의한 그림자가 존재하는 경우이다. 이들 역시 강한 검은색을 띠기 때문에 단순히 픽셀값을 사용하여 균열 인식을 하면 그림자의 테두리 부분이 모두 균열로 인식되는 현상이 발생한다. 그러나 제안된 모델을 사용한 실험 결과는 대부분 그림자를 정확히 인식하고, 암석 균열과 구분할 수 있다. 그러나, (c)의 경우에는 좌측 상단부의 그림자를 일부 검출하는 한계점도 나타났다.

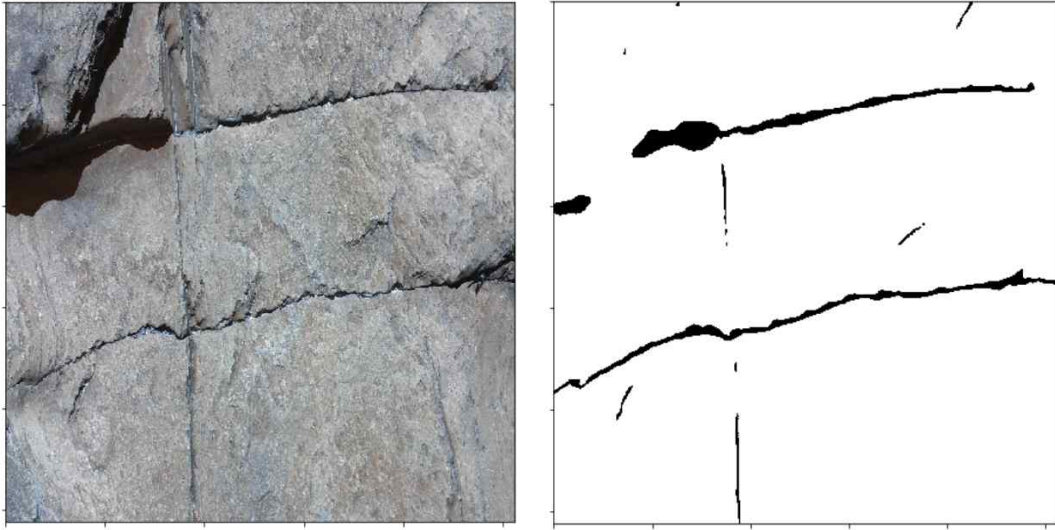


(a)

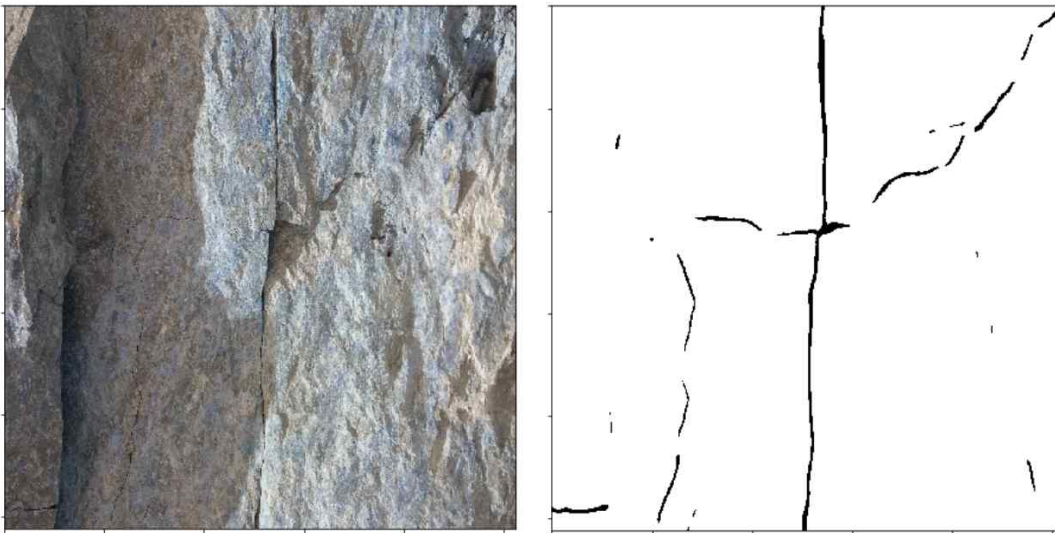


(b)

Figure 4.2 Rock fracture detection results with shadow on surface. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).

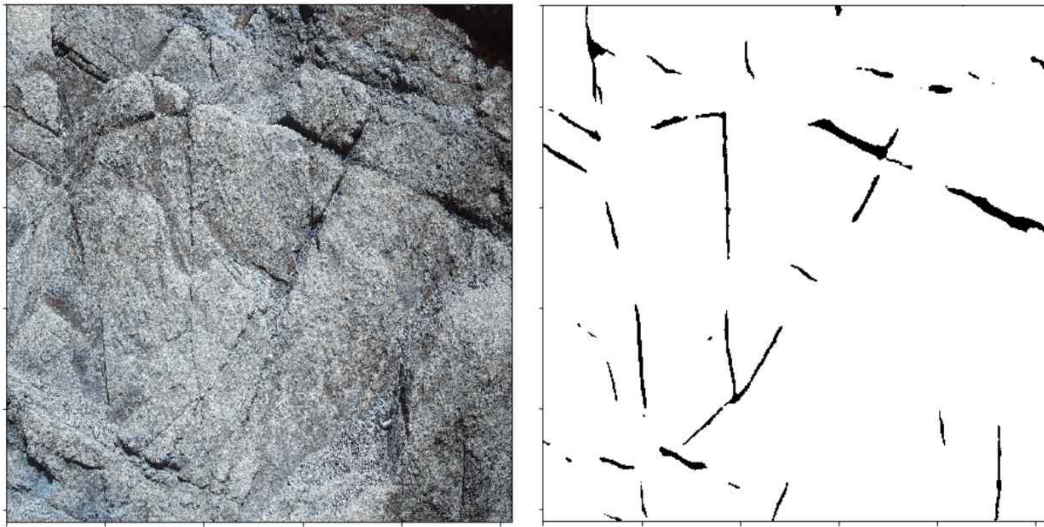


(c)



(d)

Figure 4.2 Rock fracture detection results with shadow on surface. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).



(e)



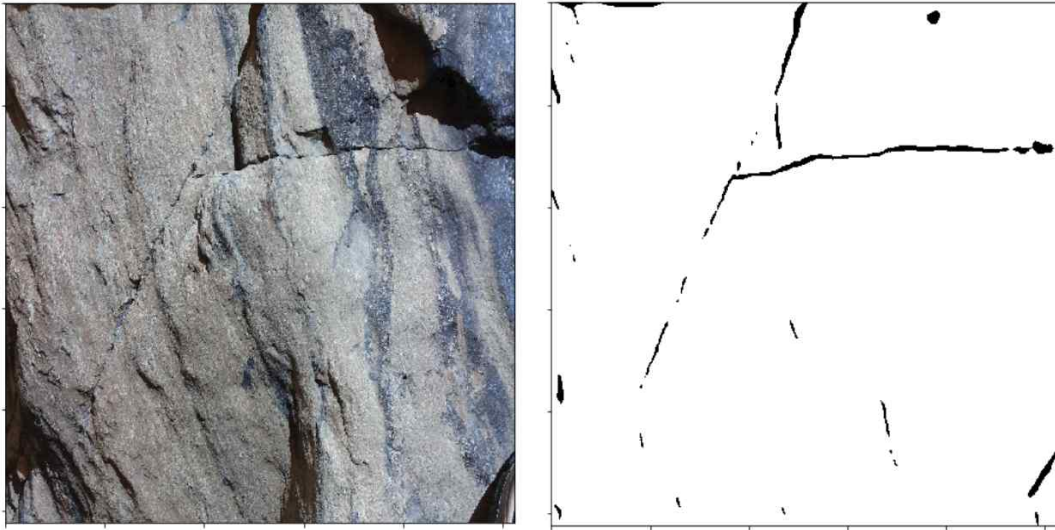
(f)

Figure 4.2 Rock fracture detection results with shadow on surface. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (continued).

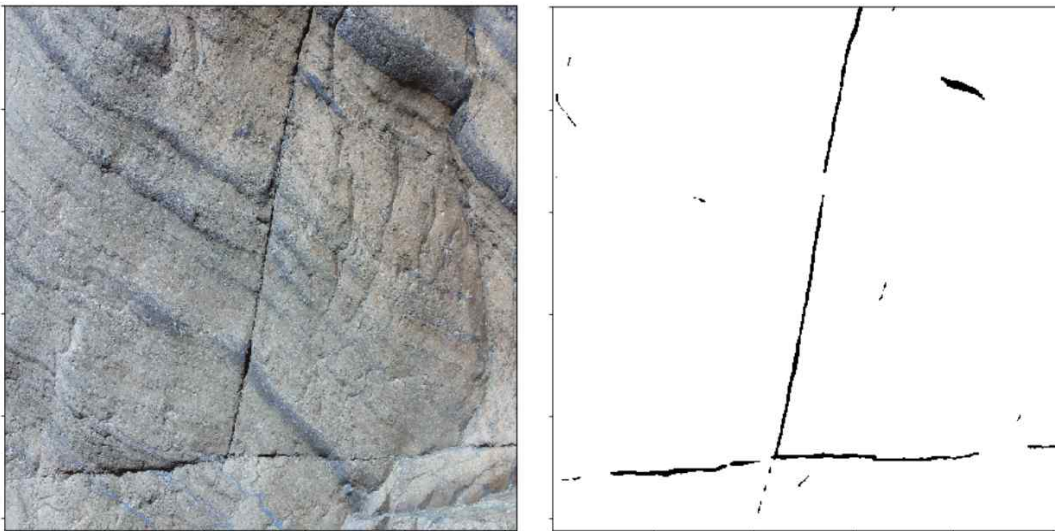
4.2 줄무늬 구조가 존재하는 사진

다음으로 암석 표면에 줄무늬 구조가 있는 경우에 대해 테스트를 수행하였다. 줄무늬 구조는 균열과 비슷한 형태를 띠며 에지 검출로도 검출될 수 있어 균열 인식을 방해한다. 본 연구에서 개발한 방법을 통해 균열 인식을 하는 경우 Fig. 4.3과 같이 표면에 줄무늬가 존재하더라도 균열만을 검출할 수 있다.

Fig. 4.3의 모든 경우에 대해서 균열 인식이 잘 이루어졌다. 특히, (c)의 경우 상부 중앙과 오른쪽 구석의 균열은 밝기값이 기질의 편리 구조에 비해 높음에도 불구하고, 인식에 성공하였다. (e)의 사진에서도 얇은 균열을 잘 찾아냈으나, 암석 요철에 의한 음영을 일부 균열로 인식하는 한계점도 보인다. 이는 학습에 사용한 암석 데이터에서 요철에 의한 음영을 포함하는 사진의 양이 충분치 못했기 때문이다. 균열과 음영을 포함한 더욱 많은 수의 사진을 학습에 사용한다면, 불필요한 음영 인식을 줄일 수 있다.

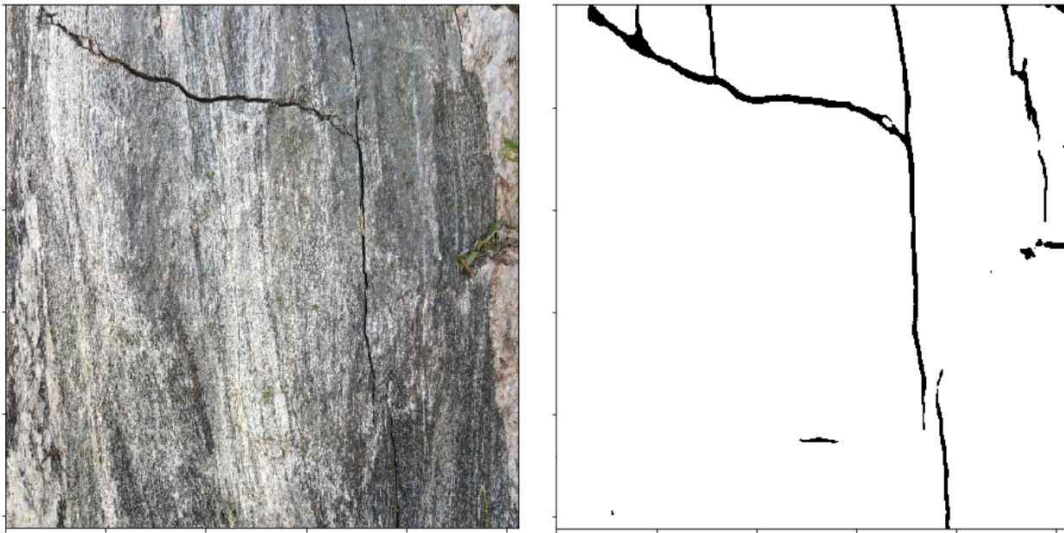


(a)

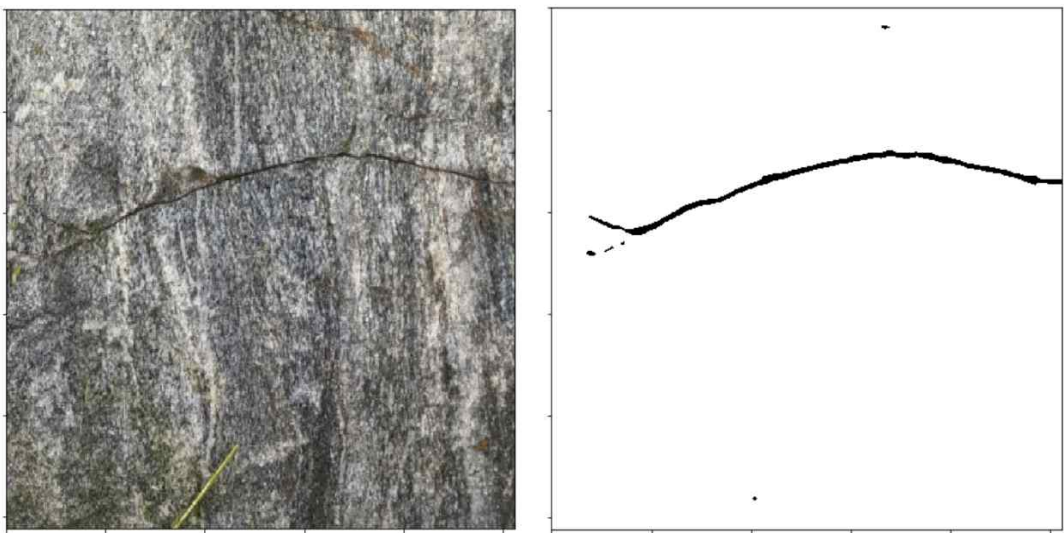


(b)

Figure 4.3 Rock fracture detection results with striped pattern. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).

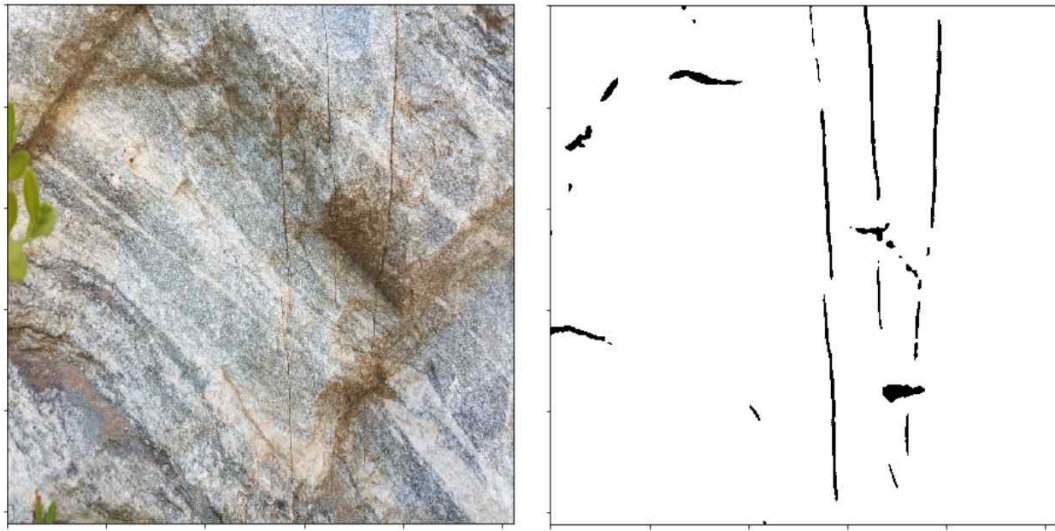


(c)



(d)

Figure 4.3 Rock fracture detection results with striped pattern. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).



(e)

Figure 4.3 Rock fracture detection results with striped pattern. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (continued).

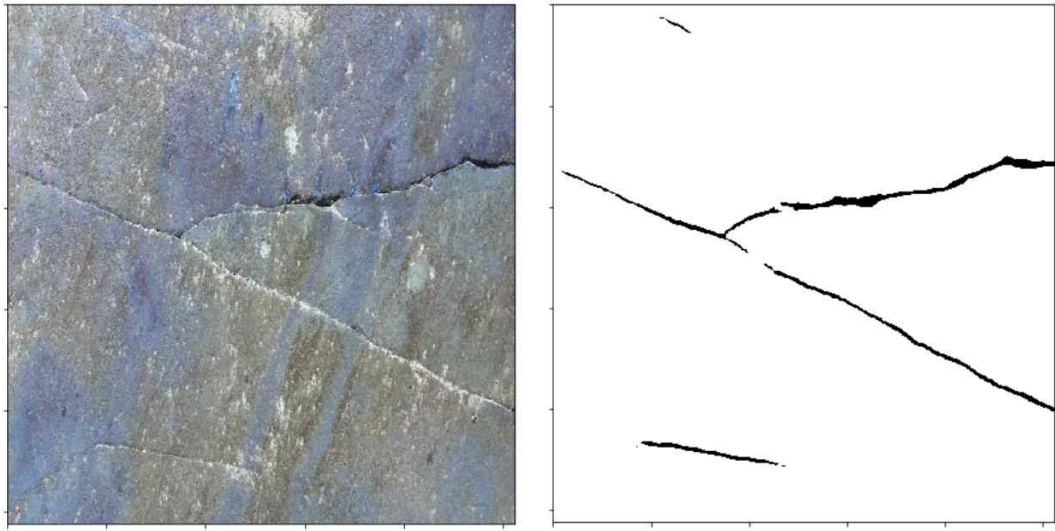
4.3 충전물이 존재하는 사진

균열이 생성된 후 시간이 흐름에 따라 균열 내부에 충전물이 생성될 수 있다. 이 경우 충전물에 의해 균열의 시각적 정보가 달라진다. 일반적으로 균열은 기질보다 어두운색을 띤다. 그러나 충전물이 존재할 때는 다양한 경우가 존재한다. 따라서, 기존의 규칙 기반(rule base)추론 방식을 사용할 경우 많은 조건을 고려하며 알고리즘을 개발해야 한다. 이에 반해 본 연구에서는 균열에 충전물이 포함된 사진도 학습 데이터로 사용함에 따라 균열 인식이 간단히 이루어진다.

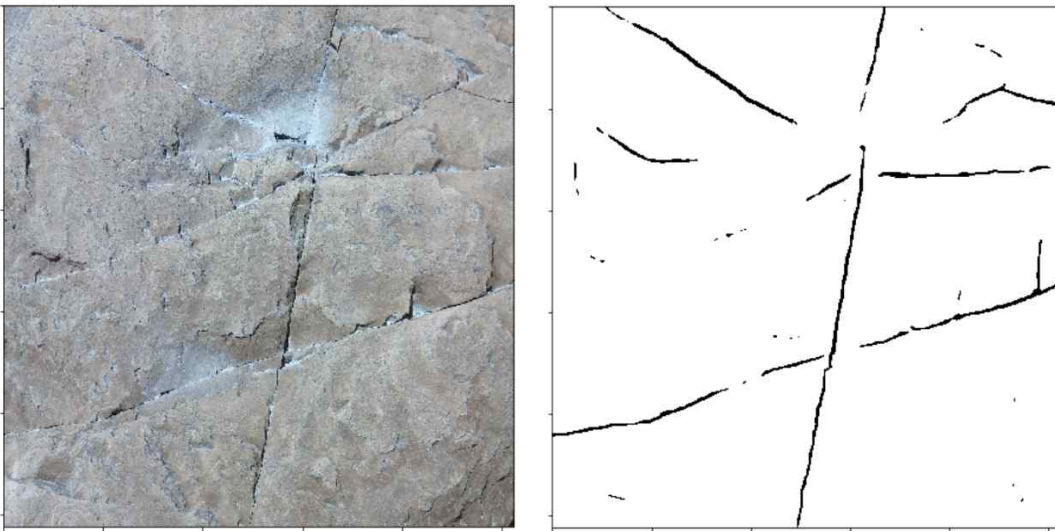
Fig. 4.4는 균열 일부에 충전물이 존재하는 균열 사진과 인식 결과를

보여준다. (a)~(c)는 균열 일부분을 유색 충전물이 가득 메꾼 경우이다. (b) 사진의 상부에는 하얀색 충전물이 균열뿐 아니라 암석 표면에도 일부 분산되어 있다. 이 경우 학습 데이터의 부족으로 인해 균열을 완벽히 검출하지 못하는 한계를 보인다. 그러나 일부 균열의 내부에 충전물이 들어있을 때는 균열 인식이 잘 이루어지는 것을 확인할 수 있다.

Fig. 4.4 (d)는 균열에 거미줄과 같은 생물의 흔적이 존재하는 경우를 보여준다. 이 경우는 거미줄이 기질과 균열에 걸쳐서 존재하기 때문에 기질과 균열의 경계 부분을 정확히 판단하기 모호한 부분이 있다. 이 경우는 학습용 데이터를 만들 때도 존재하였으며, 균열 경계 판단에 불확실성이 존재한다고 판단하였다. 결국, 장애물로 가려진 부분을 제외한 확실히 균열임을 알 수 있는 부분만을 균열로 레이블링을 하였다. 따라서 테스트 결과도 확실한 균열 부분만을 검출한다.



(a)



(b)

Figure 4.4 Rock fracture detection results with infill material. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).



(c)



(d)

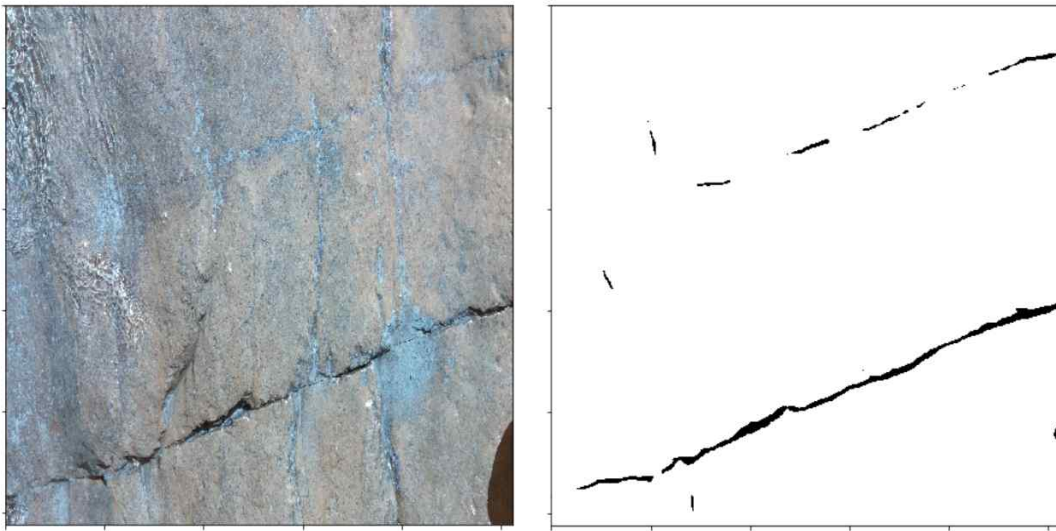
Figure 4.4 Rock fracture detection results with infill material. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (continued).

4.4 굵힌 자국이 존재하는 사진

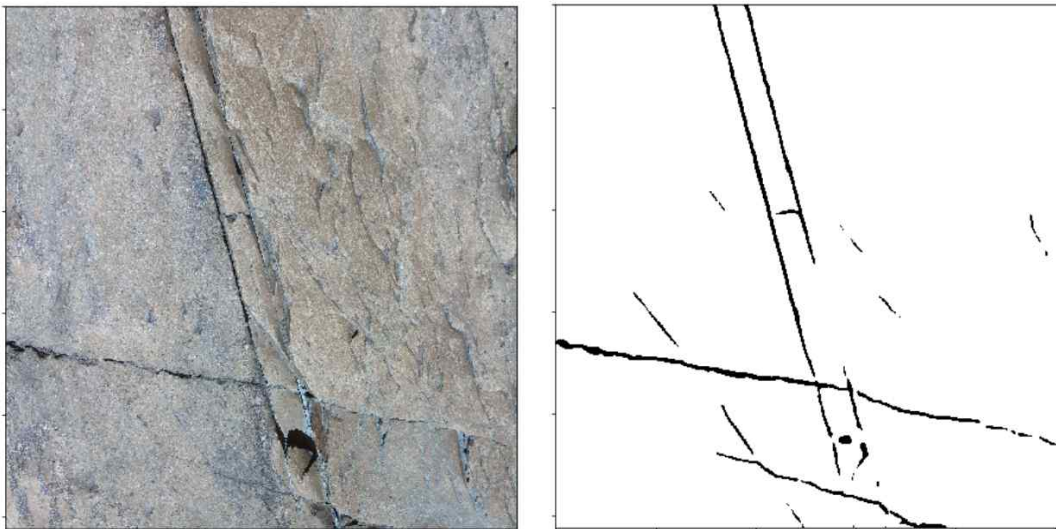
앞서 명시한 바와 같이 본 연구에서는 단순한 표면의 굵힌 흔적과 구분되는 깊이감이 있는 균열만을 인식하는 것을 목표로 하였다. 따라서 암석 표면에 단순 굵힌 자국이 포함된 사진으로 테스트를 수행하였다.

Fig. 4.5 (a)에는 표면의 굵힌 자국에 파란색의 충전물도 포함되어 있다. 하단부의 균열은 잘 검출하였으나, 파란 굵힌 자국은 일부는 균열로 인식하였으며, 나머지는 균열이 아닌 것으로 인식하였다. 상단부의 가로 방향 자국의 경우 하단부의 확실한 균열과 평행하여 균을 이루므로 균열일 확률이 높아 균열로 인식됐지만, 세로 자국의 경우는 균열이 아닌 것으로 판단하였다. Fig. 4.5 (b)에는 균열 외에도 작은 크기의 굵힌 자국들이 우측 상단부에 다수 존재한다. 이들 굵힌 자국들은 깊이감이 없으며 암석 표면에만 존재하는 흔적들이기 때문에 인식되지 않았다. 그 외에도 우측 하단부의 균열에는 충전물이 채워져 있으나 잘 검출되었다.

Fig. 4.5 (c)는 다량의 균열과 단순 자국들이 혼합되어 있다. 균열과 굵힌 자국의 경계가 모호한 일부에 대해서는 신경망 모델이 완전히 구분하지는 못하는 것을 볼 수 있다. 신경망의 추론 원리는 학습 데이터를 기반으로 하며 학습 데이터는 사람이 사진으로부터 만들었다. 신경망의 판단 근거는 전적으로 학습 데이터와 사람의 레이블링 결과를 기반으로 한다. 따라서 사람이 사진에서 육안으로 판단함에도 모호함이 존재하는 균열의 경우에는 신경망의 추론 결과에도 불확실성이 나타난다.



(a)



(b)

Figure 4.5 Rock fracture detection results with surface scratch. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).



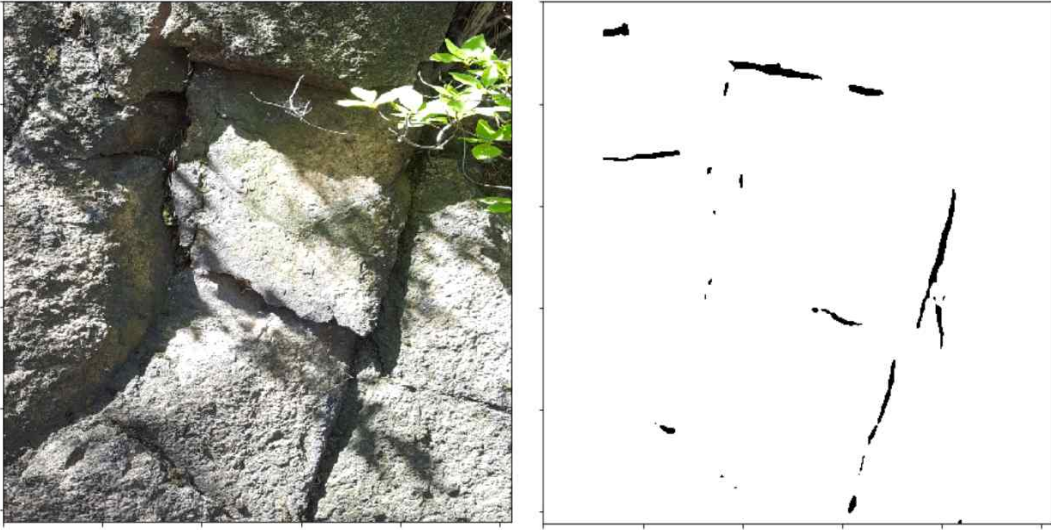
(c)

Figure 4.5 Rock fracture detection results with surface scratch. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (continued).

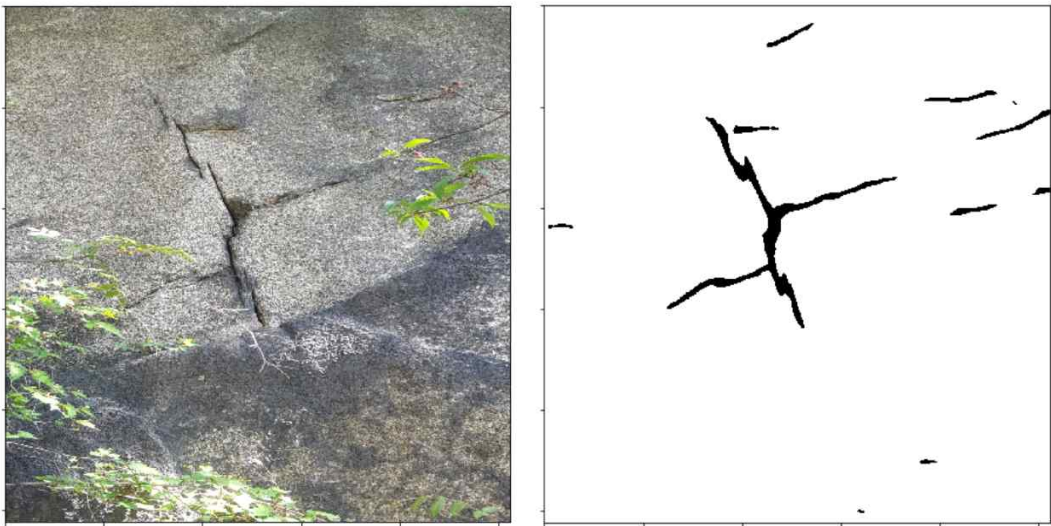
4.5 식생이 존재하는 사진

마지막으로 사진에 식생이 존재하는 경우에 대해 테스트를 수행하였다. 노출된 암석에는 풍화작용이 일어나며 그 결과 발생한 흠에 식생이 자라게 된다. 따라서 식생이 존재하는 암석의 경우 보통 풍화가 많이 일어난 암석이며, 균열이 많이 발달해 있을 확률이 높기에 강도 저하가 예상된다. 결국, 식생이 발달한 암석은 균열 인식을 통한 조사가 필요한 경우가 많기에 이들 암석에 대한 균열 인식의 필요성이 대두된다. 식생을 포함한 암석 사진을 촬영할 때 사진에 식생이 필연적으로 포함되는 경우가 있으며, 이러한 사진들을 사용하여 균열 인식 테스트를 진행하였다.

Fig. 4.6은 식생이 존재하는 경우의 균열 인식 결과를 나타낸다. (a), (b)는 사진 속에 나뭇잎과 나뭇가지가 있다. 나뭇가지는 선형성을 띠기 때문에 균열과 비슷한 특징을 가지며 에지 검출 알고리즘 사용 시 균열로 잘못 인식될 수 있다. (a)와 (b)의 좌측에 존재하는 나뭇가지는 개발된 모델을 통해 구별할 수 있었으나, (b)의 우측 상단부에 있는 나뭇가지는 균열로 잘못 인식하는 결과가 발생했다. (b)의 좌측의 나뭇가지는 많은 나뭇잎과 함께 존재하기 때문에 식생으로 분류되었으나, 우측의 나뭇가지는 적은 수의 나뭇잎을 가지며 두께, 색상이 균열과 매우 유사하기에 예측 실패가 발생한 것으로 판단된다. 한편, (a)의 암석은 균열의 발달이 거의 완료되어 암석들이 일부 날개로 분리되었다. 중앙 좌측부의 균열 좌우의 두 암석은 완전히 분리된 다른 암석으로 판단되었기 때문에 사이 부분은 균열로서 검출되지 않았다. 그러나, 판단 기준에 따라 균열로 인식돼야 할 필요성도 있는 만큼 이와 같은 모호한 경우에 대해 집중적인 신경망 학습이 추가로 필요할 것으로 보인다. (c)는 우측 하단부에 들풀이 존재하는 경우를 보여준다. 이 경우에도 들풀은 암석과 구분되었으며, 들풀이 균열의 하단부를 가리고 있는 만큼, 균열의 확실한 경계를 알 수 있는 부분만 검출되었다.

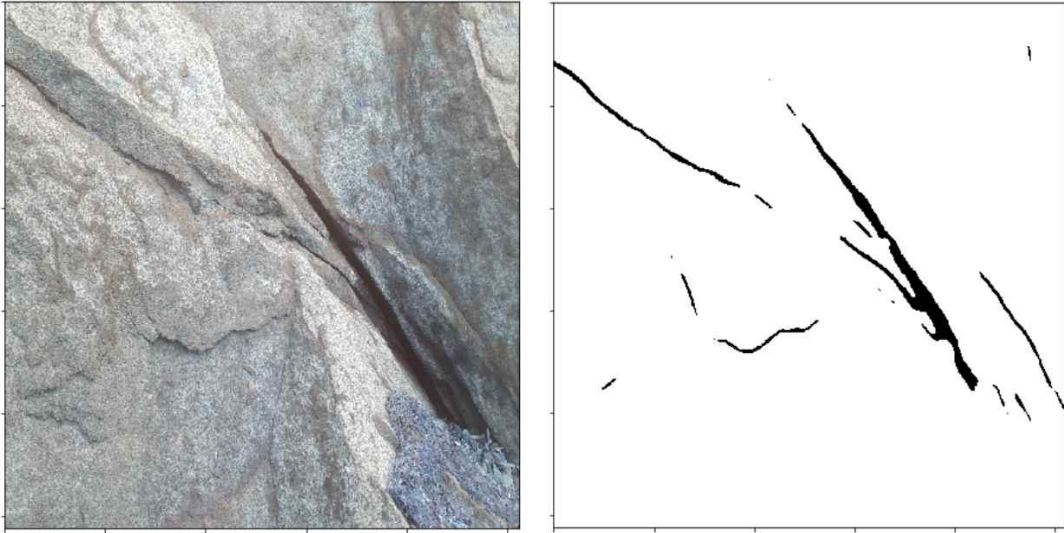


(a)



(b)

Figure 4.6 Rock fracture detection results with vegetation. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (to be continued).



(c)

Figure 4.6 Rock fracture detection results with vegetation. Raw images of rock fractures used in the test and corresponding results of fracture segmentation (continued).

4.6 모델 성능 고찰

암반 균열 인식의 주요 목표는 실제 균열 인식을 최대화하는 동시에 노이즈의 인식을 최소화 하는 것이다. 기존 연구들은 균열 인식에 저차원의 피처를 사용하였기 때문에 단순 노이즈와 구별이 잘되지 않았다. 따라서 전처리를 통해 노이즈를 제거하는 과정을 앞 단계에 추가하였다 (Wang, 2005, Wang et al., 2007, Liang, 2016). 그러나 이와 같은 방법은 노이즈 제거 정도가 높을수록 균열도 같이 제거되는 문제를 동반한다. 반면에 본 연구에서 개발한 모델은 균열 검출을 위해 대상 영역 외에 주변 영역도 함께 고려하는 등 고차원 피처를 사용하기 때문에, 부가

적으로 노이즈를 제거하지 않아도 균열 검출 과정에서 노이즈가 구별된다.

그럼에도 불구하고 일부 사진에서는 노이즈를 인식하거나 균열을 인식하지 못하는 경우가 발생하였다. 그 이유는 다음과 같다. 첫째, 모델 학습은 학습용 데이터의 레이블링에 전적으로 의존하기 때문이다. 학습 데이터를 만들 때 특정한 형상의 균열을 단순 노이즈로 판별하여 학습시켰다면, 비슷한 균열을 테스트하는 경우에 노이즈로 인식할 것이다. 균열을 레이블링한 결과도 개인별 편차를 보이기 때문에, 서로 다른 사람이 레이블링한 데이터로 학습한 모델은 다른 판단 기준을 보인다. 이를 개선하기 위해서는 학습 데이터를 만드는 과정에서 다양한 판별 기준을 고려하여, 학습 데이터의 신뢰도를 높여야 한다. 둘째, 학습 데이터가 충분하지 않기 때문이다. 다양한 종류의 균열 데이터를 수집하여 모델 학습을 수행하였지만, 테스트 데이터에 완전히 새로운 형태의 균열이나 노이즈가 발생한다면 인식 성능이 저하된다. 이는 더욱 많은 종류의 데이터를 학습에 사용하여 해결할 수 있다.

개발한 모델의 성능을 정량적으로 평가하기 위해 균열 인식 정확도를 계산하였다. 기존 암석 균열 인식 연구에서는 균열의 모호성 등을 이유로 정량적인 모델 평가가 거의 이루어지지 않았다. 그러나 암석 균열 인식 모델들을 꾸준히 개발하고 발전시키는 과정에서 모델 성능 평가는 필수적이다. 본 연구에서는 암석 균열 인식 성능 평가를 위해 테스트 데이터의 레이블과 모델 예측 결과를 비교 분석하였다(Fig. 4.7).

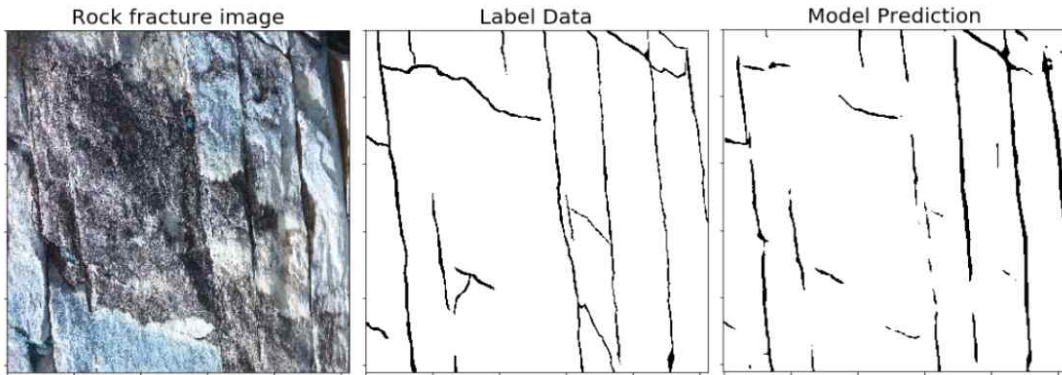


Figure 4.7 Rock fracture image, its label, and prediction results.

모델 성능 평가는 노이즈를 제거 성능과 실제 균열 검출 성능을 모두 평가할 수 있도록 Precision, Recall, F1 score를 사용하였다. 각각의 정의는 식 (12)와 같다.

$$\begin{aligned}
 Precision &= \frac{TP}{TP + FP}, \\
 Recall &= \frac{TP}{TP + FN} \\
 F1\ score &= \frac{2(Precision \times Recall)}{Precision + Recall}
 \end{aligned}
 \tag{12}$$

TP (true positive)는 균열 중 인식에 성공한 것, FP (false positive)는 균열이 아닌 것 중 균열로 인식이 된 것, FN (false negative)는 균열 중 인식되지 않은 것이다. Precision은 인식 결과 중 실제 균열의 비중을, Recall은 실제 균열이 인식된 정도를 의미한다. F1 score는 Precision과 Recall의 조화평균으로서 두 지표를 모두 고려한 수치이다. 테스트 데이터 23개 각각에 대하여 Precision, Recall, F1 score를 계산하였고, 그 값을 평균한 결과를 Table 4.1에 표기하였다.

Table 4.1 Rock fracture detection performance of proposed model

Precision	Recall	F1 score
0.508	0.633	0.564

5. 결론

본 연구에서는 딥러닝 기반의 합성곱 신경망을 사용하여 사진측량법을 통해 얻은 암반 사진으로부터 균열을 자동으로 인식하는 모델을 개발하였다. 사진 측량법은 접근하기 힘든 지역을 목적으로 하므로 원거리에서 촬영된 큰 사이즈의 사진이 얻어지며 각종 장애물을 포함할 확률이 높다. 본 연구에서 개발한 모델은 기존 연구들과 달리 다양한 노이즈가 존재해도 균열 검출을 효과적으로 수행하므로 사진측량법을 통한 균열 검출에 사용하기 적합하다. 본 연구의 주요 결론은 다음과 같다.

(1) 암반 균열 인식에 최적화된 피처를 스스로 찾아낼 수 있는 합성곱 신경망을 개발하였다. 이를 통해 과거 이미지 처리를 통한 균열 검출 기법처럼 사람이 시행착오를 통해 직접 피처를 추출하는 과정에 비해 빠른 모델 개발과 성능 향상이 가능하다. 또한, 기존 연구들과 달리 커널 연산을 반복적으로 수행하는 합성곱 신경망을 사용했기 때문에 단순 에지 정보가 아닌 고차원 피처를 균열 인식에 사용하였다.

(2) 기존 문헌들은 균열 외의 노이즈가 거의 없고 균열 형태가 단순한 정형화된 균열 이미지를 사용하였으나, 본 연구에서는 균열 인식을 방해할 수 있는 다양한 조건들이 추가된 경우에서의 균열 인식을 목표로 하였다. 그 결과 새로 개발된 모델은 그림자, 줄무늬, 충전물, 굽힌 자국, 식생 등이 존재해도 효과적으로 균열을 인식하였다.

(3) 모델 개발 과정에서 모델 내부 파라미터들을 결정하는 과정에서 사용되지 않은 새로운 균열 데이터들로 테스트를 수행하였다. 학습 데이터와 테스트 데이터를 분리하지 않은 기존 연구 방법을 따르면 새로운 데이터에 대한 성능 보장이 힘든 반면, 본 연구에서 개발된 모델은 새로운 데이터에 대해서도 높은 균열 인식 성능이 예상된다.

(4) 특이한 경우의 비정형 데이터들과 균열 여부가 모호한 일부 데이터에 대해서는 예측이 실패하였다. 이는 테스트에 사용한 균열 데이터와 비슷한 특성을 갖는 학습 데이터가 부족하기 때문이다. 추후에 학습에 사용한 데이터의 수와 종류를 증가시킨 후 추가로 신경망 학습을 진행한다면 보다 많은 종류의 균열을 인식할 수 있을 것이다.

참 고 문 헌

- Alqahtani, N., Armstrong, R. T., Mostaghimi, P., 2018, Deep learning convolutional neural networks to predict porous media properties, SPE Asia Pacific oil and gas conference and exhibition, Society of Petroleum Engineers.
- Chen, L. C., Papandreou, G., Schroff, F., Adam, H., 2017, Rethinking atrous convolution for semantic image segmentation, arXiv preprint arXiv:1706.05587.
- Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2018, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE transactions on pattern analysis and machine intelligence, Vol. 40, No. 4, pp. 834-848.
- Cheng, G., Guo, W., 2017, Rock images classification by using deep convolution neural network, In Journal of Physics: Conference Series, Vol. 887, No. 1, p. 012089.
- Dalal, N., Triggs, B., 2005, Histograms of oriented gradients for human detection. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, pp. 886 - 893.
- Deb, D., Hariharan, S., Rao, U. M., Ryu, C. H., 2008, Automatic detection and analysis of discontinuity geometry of rock mass from digital images, Computers & Geosciences, Vol. 34, No. 2, pp. 115-126.
- Ferreira, A., Giraldi, G., 2017, Convolutional neural network approaches to granite tiles classification, Expert Systems with Applications, Vol. 84, pp. 1-11.

- He, C., Wang, W., 2010, A PCNN-based edge detection algorithm for rock fracture images, 2010 Symposium on Photonics and Optoelectronics, pp. 1-4.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861.
- Ioffe, S., Szegedy, C., 2015, Batch normalization: Accelerating deep network training by reducing internal covariate shift, arXiv preprint arXiv:1502.03167.
- Karimpouli, S., Tahmasebi, P., 2019, Image-based velocity estimation of rock using Convolutional Neural Networks, Neural Networks, Vol. 111, pp. 89-97.
- Karimpouli, S., Tahmesbi, P., 2019, Segmentation of digital rock images using deep convolutional autoencoder networks, Computers & Geosciences, Vol. 126, pp. 142-150.
- Kingma, D. P., Ba, J. L., 2014, Adam: A Method for Stochastic Optimization Adam optimizer, arXiv preprint arXiv:1412.6980v9.
- Krähenbühl, P., Koltun, V., 2011, Efficient inference in fully connected crfs with gaussian edge potentials, Advances in neural information processing systems, pp. 109-117.
- Liang, Y., 2016, Rock fracture skeleton tracing by image processing and quantitative analysis by geometry features, Journal of Geophysics and Engineering, Vol. 13, No. 3, pp. 273-284.
- Long, J., Shelhamer, E., Darrell, T., 2015, Fully convolutional networks for semantic segmentation, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431-3440.
- Lowe, D. G., 1999, Object recognition from local scale-invariant

- features. Proceedings of the International Conference on ComputerVision, Vol. 99, No. 2, pp. 1150–1157.
- McCarter, M.K., 1976, Application of plane photogrammetry to open pit mapping. US Symposium on Rock mechanics, Snowbird, Society of Mining Engineers of AIME, pp. 93–98.
- Mohebbi, M., Yarahmadi Bafghi, A. R., Fatehi Marji, M., Gholamnejad, J., 2017, Rock mass structural data analysis using image processing techniques (Case study: Choghart iron ore mine northern slopes), Journal of Mining and Environment, Vol. 8, No. 1, pp. 61–74.
- Nair, V., Hinton, G. E., 2010, Rectified linear units improve restricted boltzmann machines, Proceedings of the 27th international conference on machine learning, pp. 807–814.
- Nishad, P. M., 2013, Various colour spaces and colour space conversion, Journal of Global Research in Computer Science, Vol 4, No. 1, pp. 44–48.
- Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar romeny, B. M., Zimmerman, J. B., Zuiderveld, K., 1987, Adaptive histogram equalization and its variations, Computer vision, graphics, and image processing, Vol. 39, No. 3, pp. 355–368.
- Ronneberger, O., Fischer, P., Brox, T., 2015, U-net: Convolutional networks for biomedical image segmentation, International Conference on Medical image computing and computer-assisted intervention, pp. 234–241.
- Scherer, D., Müller, A., Behnke, S., 2010, Evaluation of pooling operations in convolutional architectures for object recognition, International conference on artificial neural networks, pp. 92–101.

- Seetal, P., Natarajan, N., 2010, Image segmentation for rock fractures based on arma model, *International Journal of Engineering Science and Technology*, Vol. 2, No. 5, pp. 1155-1159.
- Simard, P. Y., Steinkraus, D., Platt, J. C., 2003, Best practices for convolutional neural networks applied to visual document analysis, *Icdar*, Vol. 3, No. 2003.
- Wada, Ketaro, 2016, labelme :Image Polygonal Annotation with Python, GitHub repository, <https://github.com/wkentaro/labelme>
- Wang, W., 2005, An edge based segmentation algorithm for rock fracture tracing, *International Conference on Computer Graphics, Imaging and Visualization*, pp. 43-48.
- Wang, W., Liao, H., Huang, Y., 2007, Rock fracture tracing based on image processing and SVM, *Third International Conference on Natural Computation*, Vol. 1, pp. 632-635.
- Wang, W. X., Li, W. S., Yu, X, 2012, Fractional differential algorithms for rock fracture images, *The Imaging Science Journal*, Vol. 60, No. 2, pp. 103-111.
- Wei, Z.Q., Egger, P., Descoedres, F., 1995, Permeability predictions for jointed rock mass, *International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts*, Vol. 32, No. 3, pp. 251-261.

Abstract

Automated Rock Fracture Detection Algorithm with Convolutional Neural Networks

Hoon Byun

Department of Energy Systems Engineering

The Graduate School

Seoul National University

Detection of rock joint and fracture is important because they have a huge influence on rock mass strength. Photogrammetry technique, especially, has been used for decades due to its simplicity and economic feasibility. Although joints are easy to detected since it has linearity, fractures has irregularity which leads to difficulties in detection and lack of relevant studies. Additionally, previous researches used photographs without various types of noise such as shadow, infill material and vegetation. These kinds of noise reduce the accuracy of conventional algorithms. However, it can be included in the photographs under certain circumstances.

In this study, a new algorithm based on convolutional neural networks, which can detect rock fracture from rock images with

many kinds of noise, is presented. Furthermore, previous models were evaluated with the same image used in model construction stage. The model performance, therefore, is guaranteed only for that specific data. On the contrary, new rock images are used when testing the model, which shows the data-independent performance of proposed model. As a result, the developed model in this study can detect rock fracture from photographs quickly and consistently, and demonstrate high performance for irregular fractures.

keywords : fractures, image detection, convolutional neural networks

Student Number : 2015-22944