(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년06월30일
(11) 등록번호 10-2550707
(24) 등록일자 2023년06월28일

(51) 국제특허분류(Int. Cl.)
G01B 11/16 (2006.01) *G01B 11/02* (2006.01)
G06N 3/04 (2023.01) *G06N 3/08* (2023.01)
G06Q 50/10 (2012.01) *G06T 7/00* (2017.01)
G06T 7/10 (2021.01) *G06T 7/194* (2017.01)
G06T 7/62 (2017.01) *G06T 7/66* (2017.01)
G08B 21/10 (2014.01)
(52) CPC특허분류
G01B 11/16 (2013.01)
G01B 11/02 (2013.01)
(21) 출원번호 10-2022-0130678
(22) 출원일자 2022년10월12일
심사청구일자 2022년10월12일
(56) 선행기술조사문헌
CN115063963 A*
(뒷면에 계속)

(73) 특허권자
부경대학교 산학협력단
부산광역시 남구 용소로 45, 부경대학교내 (대연동)
(72) 발명자
김윤태
부산광역시 남구 분포로 113, LG메토로시티 218동 903호 (용호동)
하용수
부산광역시 남구 용소로 45 ,환경연구동313호(대연동)
팜 밍 부옹
부산광역시 남구 용소로21번길 83-1, 202호
(74) 대리인
네이트특허법인

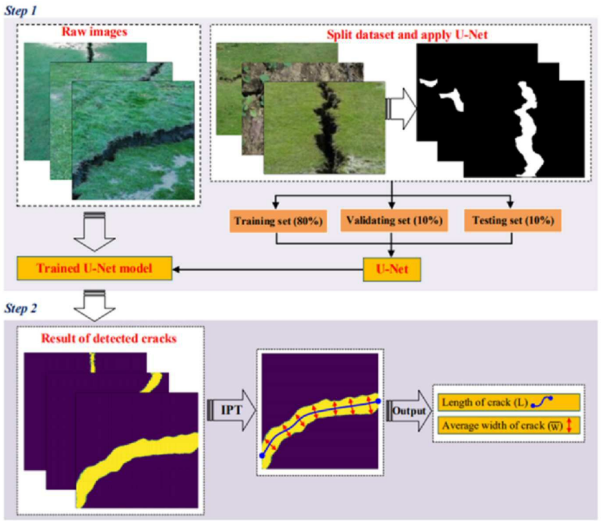
전체 청구항 수 : 총 18 항 심사관 : 홍정훈

(54) 발명의 명칭 **딥러닝 기반 경사지 균열 감지 방법과 이를 이용한 산사태 조기 감지 방법 및 그 장치**

(57) 요 약

본 발명은, 디지털 영상 정보를 분석하여 산사태의 징조가 될 수 있는 경사지의 균열 및/또는 토석류 거동을 감지 및 계산함으로써, 산사태를 조기에 감지 및/또는 경고할 수 있도록 하는 딥러닝 기반 경사지 균열 감지 방법과 이를 이용한 산사태 조기 감지 방법 및 그 장치에 관한 것이다. 본 발명의 일 실시예에 따른 딥러닝 기반 경사지 균열 감지 방법은, 이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계; 상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 및 상기 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계;를 포함할 수 있다. 이때, 상기 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성할 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06N 3/047 (2023.01)
G06N 3/08 (2023.01)
G06Q 50/10 (2015.01)
G06T 7/001 (2013.01)
G06T 7/10 (2021.01)
G06T 7/194 (2017.01)
G06T 7/62 (2017.01)
G06T 7/66 (2017.01)
G08B 21/10 (2013.01)

(56) 선행기술조사문헌
JP2022520994 A*
JP3636998 B2*
KR102254773 B1*
US20170343481 A1*
*는 심사관에 의하여 인용된 문헌

이 발명을 지원한 국가연구개발사업

과제고유번호	1315001786
과제번호	20019512
부처명	행정안전부
과제관리(전문)기관명	한국산업기술평가관리원
연구사업명	지역맞춤형재난안전문제해결기술개발지원(R&D)
연구과제명	AI 기반 주거지 옹벽안전관리 플랫폼 구축 및 실증
기 여 율	1/1
과제수행기관명	(재)부산테크노파크
연구기간	2022.04.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계;

상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 및

상기 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계;를 구비하고,

상기 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성하며,

상기 균열 크기 산출 단계에는 상기 균열 감지 이미지를 이미지 처리하여 식별된 균열 형상으로부터 균열의 길이와 너비를 계산하며,

상기 균열 형상을 설정된 수평 또는 수직 픽셀 수의 단위로 복수의 세그먼트로 분할하고, 기준선과 균열의 경계의 두 교차점의 중심을 각각의 세그먼트의 중심점으로 하고, 해당 세그먼트의 중심점과 이전 세그먼트의 중심점 사이의 거리를 해당 세그먼트의 중심선으로 하고, 모든 세그먼트의 중심선의 합이 상기 균열의 길이로 계산되는 딥러닝 기반 경사지 균열 감지 방법.

청구항 2

제1항에 있어서,

상기 균열 감지 이미지가, 상기 입력 이미지를 1개 채널의 회색 이미지로 변환하고, 상기 회색 이미지의 각각의 픽셀이 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 큰 경우 균열에 해당하는 값을 할당하고, 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 작거나 같은 경우 배경에 해당하는 값을 할당하여 생성되는 딥러닝 기반 경사지 균열 감지 방법.

청구항 3

제1항에 있어서,

상기 U-Net 모델이,

2개 이상의 컨볼류션 블록을 포함하여 다운 샘플링하는 수축 경로, 및

2개 이상의 컨볼류션 블록을 포함하여 업 샘플링하는 확장 경로를 포함하고,

상기 컨볼류션 블록이 컨볼류션(Convolution), 배치 정규화(Batch Normalization, BN), 및 ReLU (Rectified Linear Unit) 연산을 포함하고,

상기 수축 경로는 최대 풀링(Maxpooling) 작업을 포함하나, 상기 확장 경로는 최대 풀링(Maxpooling) 작업을 포함하지 않는 딥러닝 기반 경사지 균열 감지 방법.

청구항 4

삭제

청구항 5

삭제

청구항 6

제1항에 있어서,

상기 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 0~45도이면 균열 감지 이미지에서 수직선이 상기 기준선이 되고,

상기 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 45~90도이면 균열 감지 이미지에서 수평선이 상기 기준선이 되는 딥러닝 기반 경사지 균열 감지 방법.

청구항 7

제1항에 있어서,

해당 균열 세그먼트에서 상기 중심선에 수직인 선과 균열 경계의 교차하는 두 지점들 사이의 거리가 해당 균열 세그먼트의 너비가 되고,

전체 균열의 너비는 각각의 균열 세그먼트 너비들의 평균인 딥러닝 기반 경사지 균열 감지 방법.

청구항 8

제1항에 있어서,

픽셀 단위로 계산된 상기 균열의 길이와 너비에 ㎜ 단위 변환 계수를 곱하여 ㎜ 단위의 상기 균열의 길이와 너비를 계산하며,

상기 ㎜ 단위 변환 계수는 ㎜ 단위의 설정된 면적에 포함되는 가로 픽셀수 X 세로 픽셀수의 값으로 나눈 값의 제곱근인 딥러닝 기반 경사지 균열 감지 방법.

청구항 9

제1항 내지 제3항 및 제6항 내지 제8항 중의 적어도 어느 하나의 항의 딥러닝 기반 경사지 균열 감지 방법을 구비하는 산사태 조기 감지 방법.

청구항 10

제9항에 있어서,

상기 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단하는 단계를 더 구비하는 산사태 조기 감지 방법.

청구항 11

제10항에 있어서,

상기 산사태 균열의 크기는 상기 균열의 길이와 너비를 포함하고,

상기 균열의 길이와 상기 균열의 너비의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계 중의 어느 하나로 판단하는 산사태 조기 감지 방법.

청구항 12

제10항에 있어서,

제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 어느 하나로 판단하는 산사태 조기 감지 방법.

청구항 13

제12항에 있어서,

상기 제1 파라미터는 상기 균열의 크기로부터 결정되는 값이며,

상기 제2 파라미터는 토석류의 움직임 정보로부터 결정되는 값인 산사태 조기 감지 방법.

청구항 14

제13항에 있어서,

상기 주의 단계, 상기 경고 단계, 및 상기 대피 단계가

상기 제1 파라미터와 상기 제2 파라미터의 관계가 동일한 크기의 음의 기울기를 서로 다른 직선들에 의하여 결정되는 산사태 조기 감지 방법.

청구항 15

제10항에 있어서,

상기 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성하는 경고 단계를 더 구비하는 산사태 조기 감지 방법.

청구항 16

제13항에 있어서,

상기 토석류의 움직임 정보는 상기 토석류의 속도 및 토석류의 크기에 대한 정보를 포함하는 산사태 조기 감지 방법.

청구항 17

제16항에 있어서,

상기 토석류의 움직임은 YOLO 모델의 딥러닝에 의하여 식별되고,

상기 토석류의 속도 및 크기는 이미지 처리에 의하여 계산되는 산사태 조기 감지 방법.

청구항 18

제17항에 있어서,

상기 토석류의 크기는 시간당 부피로 계산될 수 있으며, 이미지 처리에 의하여 계산되는 토석류의 속도, 폭, 및 깊이 정보에 의하여 계산되는 산사태 조기 감지 방법.

청구항 19

이미지 데이터 세트의 입력 이미지를 입력받는 영상 입력부;

상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하고, 상기 균열 감지 이미지에서 균열의 크기를 계산하는 영상 분석부;

상기 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단하는 산사태 위험 판단부; 및

상기 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성하는 경고부;를 구비하고,

상기 산사태 균열의 감지는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성하여 이루어지며,

상기 균열의 크기는 상기 균열 감지 이미지를 이미지 처리하여 식별된 균열 형상으로부터 균열의 길이와 너비를 계산하여 산출되며,

상기 균열 형상을 설정된 수평 또는 수직 픽셀 수의 단위로 복수의 세그먼트로 분할하고, 기준선과 균열의 경계의 두 교차점의 중심을 각각의 세그먼트의 중심점으로 하고, 해당 세그먼트의 중심점과 이전 세그먼트의 중심점 사이의 거리를 해당 세그먼트의 중심선으로 하고, 모든 세그먼트의 중심선의 합이 상기 균열의 길이로 계산되는 산사태 조기 감지 장치.

청구항 20

제19항에 있어서,

제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 판단하고,

상기 제1 파라미터는 상기 균열의 크기로부터 결정되는 값이며,

상기 제2 파라미터는 토석류의 움직임 정보로부터 결정되는 값인 산사태 조기 감지 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 딥러닝 기반 경사지 균열 감지 방법과 이를 이용한 산사태 조기 감지 방법 및 그 장치에 관한 것으로, 보다 상세하게는 디지털 영상 정보를 분석하여 산사태의 징조가 될 수 있는 경사지의 균열 및/또는 토석류 거동을 감지 및 계산하여 산사태를 조기에 감지 및/또는 경고할 수 있도록 하는 딥러닝 기반 경사지 균열 감지 방법과 이를 이용한 산사태 조기 감지 방법 및 그 장치에 관한 것이다.

배 경 기 술

[0002] 산사태(landslide)는 호우, 지진, 화산폭발, 눈사태 등의 원인에 의하여 산지의 급사면을 구성하는 물질이 하부로 급격히 이동하는 현상을 말한다. 급사면이 자중으로 붕괴되는 사면붕괴, 사면에서 전단강도보다 약한 면을 따라 활동이 일어나는 활동파괴, 폭우시 급사면의 느슨한 토사가 물과 함께 흘러내리는 토석류 등을 포함한 급격한 사면파괴로 나눌 수 있다. 무분별한 개발로 산의 지반이 약해지는 것이 원인이 되어 산사태가 발생하는 경우도 있다. 또 여러 가지 자연재해로 지반이 약해져 붕괴되는 경우도 있다.

[0003] 종래의 기술에 따른 토석류 산사태 모니터링 시스템에서는, 토석류가 발생할 것으로 예상되는 계곡부를 중심으로 소정의 간격을 두면서 횡방향으로 여러 개의 와이어가 설치되어 직접적으로 토석류의 이동 여부와 토석류에 의한 산사태의 발생 위치를 측정할 수 있다.

[0004] 종래의 기술에 따른 토석류 산사태 모니터링 시스템은, 토사층의 이동뿐만 아니라 토석류의 움직임 및 속도를 계측하여 산사태를 보다 신속하고 정확하게 예측 및 감지할 수 있고, 이에 따라, 사전에 산사태 발생을 미리 감지하여 피해를 최소화할 수 있다.

[0005] 하지만, 종래의 기술에 따른 토석류 산사태 모니터링 시스템은, 토석류의 이동 여부와 토석류에 의한 산사태의 발생위치를 검출하기 위해서 와이어나 광원을 계곡부들 사이에 일일이 설치해야 하기 때문에 설치비용 및 시간이 증가하는 문제점이 있다.

[0006] 한편, 종래 기술에 따른 딥러닝 기반 산사태 분석 감지 방법은, 감시 대상에 대한 딥러닝 학습에 따라 CNN(Convolution Neural Network) 알고리즘의 흑백 사물 인식 정확도를 고려하여 현장감시에 필요한 수준의 감지 정확도로 산사태 영상을 정확하게 분석할 수 있다. 이 경우, 산사태 이벤트의 오감지율을 최소화할 수 있으며, 또한, 분산형 구조를 갖는 산사태 분산 감지 방법을 구현함으로써 대규모 하드웨어 자원이 필요 없이 운영비용을 절감할 수 있다.

[0007] 하지만, 종래 기술에 따른 딥러닝 기반 산사태 분석 감지 방법은 산사태의 진행 정도를 정확하게 모니터링 하지 못하는 한계가 있다.

선행기술문헌

특허문헌

[0008] (특허문헌 0001) 한국등록특허 10-0814470 (2008.03.11)

(특허문헌 0002) 한국등록특허 10-2305468 (2021.09.16)

발명의 내용

해결하려는 과제

[0009] 본 발명의 목적은, 디지털 영상 정보를 분석하여 산사태의 징조가 될 수 있는 경사지의 균열 및/또는 토석류 거동을 감지 및 계산함으로써, 산사태를 조기에 감지 및/또는 경고할 수 있도록 하는 딥러닝 기반 경사지 균열 감지 방법과 이를 이용한 산사태 조기 감지 방법 및 그 장치를 제공하는 것이다.

과제의 해결 수단

[0010] 본 발명의 일 실시예에 따른 딥러닝 기반 경사지 균열 감지 방법은, 이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계; 상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 및 상기 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계;를 포함할 수 있다.

[0011] 이때, 상기 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성할 수 있다.

[0012] 상기 균열 감지 이미지가, 상기 입력 이미지를 1개 채널의 회색 이미지로 변환하고, 상기 회색 이미지의 각각의 픽셀이 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 큰 경우 균열에 해당하는 값을 할당하고, 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 작거나 같은 경우 배경에 해당하는 값을 할당하여 생성될 수 있다.

[0013] 상기 U-Net 모델이, 2개 이상의 컨볼류션 블록을 포함하여 다운 샘플링하는 수축 경로, 및 2개 이상의 컨볼류션 블록을 포함하여 업 샘플링하는 확장 경로를 포함하고, 상기 컨볼류션 블록이 컨볼류션(Convolution), 배치 정규화(Batch Normalization, BN), 및 ReLU (Rectified Linear Unit) 연산을 포함하고, 상기 수축 경로는 최대 풀링(Maxpooling) 작업을 포함하나, 상기 확장 경로는 최대 풀링(Maxpooling) 작업을 포함하지 않을 수 있다.

[0014] 상기 균열 크기 산출 단계에, 상기 균열 감지 이미지를 이미지 처리하여 식별된 균열 형상으로부터 균열의 길이와 너비를 계산할 수 있다.

[0015] 상기 균열 형상을 설정된 수평 또는 수직 픽셀 수의 단위로 복수의 세그먼트로 분할하고, 기준선과 균열의 경계의 두 교차점의 중심을 각각의 세그먼트의 중심점으로 하고, 해당 세그먼트의 중심점과 이전 세그먼트의 중심점 사이의 거리를 해당 세그먼트의 중심선으로 하고, 모든 세그먼트의 중심선의 합이 상기 균열의 길이로 계산될 수 있다.

[0016] 상기 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 0~45도이면 균열 감지 이미지에서 수직선이 상기 기준선이 되고, 상기 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 45~90도이면 균열 감지 이미지에서 수평선이 상기 기준선이 될 수 있다.

[0017] 해당 균열 세그먼트에서 상기 중심선에 수직인 선과 균열 경계의 교차하는 두 지점들 사이의 거리가 해당 균열 세그먼트의 너비가 되고, 전체 균열의 너비는 각각의 균열 세그먼트 너비들의 평균이 될 수 있다.

[0018] 픽셀 단위로 계산된 상기 균열의 길이와 너비에 ㎜ 단위 변환 계수를 곱하여 ㎜ 단위의 상기 균열의 길이와 너비를 계산하며, 상기 ㎜ 단위 변환 계수는 ㎜ 단위의 설정된 면적에 포함되는 가로 픽셀수 X 세로 픽셀수의 값으로 나눈 값의 제곱근이 될 수 있다.

[0019] 본 발명의 일 실시예에 따른 산사태 조기 감지 방법은, 이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계; 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계; 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단하는 단계; 및 상기 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성하는 경고 단계를 포함할 수 있다.

[0020] 이때, 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성할 수 있다.

[0021] 상기 산사태 균열의 크기는 상기 균열의 길이와 너비를 포함하고, 상기 균열의 길이와 상기 균열의 너비의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계 중의 어느 하나로 판단할 수 있다.

[0022] 제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 어느 하나로 판단할 수 있다.

[0023] 상기 제1 파라미터는 상기 균열의 크기로부터 결정되는 값이며, 상기 제2 파라미터는 토석류의 움직임 정보로부터 결정되는 값이 될 수 있다.

[0024] 상기 주의 단계, 상기 경고 단계, 및 상기 대피 단계가 상기 제1 파라미터와 상기 제2 파라미터의 관계가 동일

한 크기의 음의 기울기를 서로 다른 직선들에 의하여 결정될 수 있다.

[0025] 상기 토석류의 움직임 정보는 상기 토석류의 속도 및 토석류의 크기에 대한 정보를 포함할 수 있다.

[0026] 상기 토석류의 움직임은 YOLO 모델의 딥러닝에 의하여 식별되고, 상기 토석류의 속도 및 크기는 이미지 처리에 의하여 계산될 수 있다.

[0027] 상기 토석류의 크기는 시간당 부피로 계산될 수 있으며, 이미지 처리에 의하여 계산되는 토석류의 속도, 폭, 및 깊이 정보에 의하여 계산될 수 있다.

[0028] 본 발명의 일 실시예에 따른 산사태 조기 감지 장치는, 이미지 데이터 세트의 입력 이미지를 입력받는 영상 입력부; 상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하고, 상기 균열 감지 이미지에서 균열의 크기를 계산하는 영상 분석부; 상기 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단하는 산사태 위험 판단부; 및 상기 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성하는 경고부;를 포함할 수 있다.

[0029] 이때, 상기 산사태 균열의 감지는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성하여 이루어질 수 있다.

[0030] 제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 판단할 수 있다. 이때, 상기 제1 파라미터는 상기 균열의 크기로부터 결정되는 값이 될 수 있으며, 상기 제2 파라미터는 토석류의 움직임 정보로부터 결정되는 값이 될 수 있다.

발명의 효과

[0031] 본 발명에 따르면, 디지털 영상 정보를 분석하여 산사태의 징조가 될 수 있는 경사지의 균열 및/또는 토석류 거동을 감지 및 계산함으로써, 산사태를 조기에 감지 및/또는 경고할 수 있다.

[0032] 또한, 시간에 따른 산사태 진행 상태를 모니터링할 수 있으며, 산사태 진행상태에 따라 단계별 경보가 가능하다.

도면의 간단한 설명

[0033] 도 1은 본 발명의 일 실시예에 따른 딥러닝 기반 경사지 균열 감지 및 균열 크기 측정 방법을 개략적으로 보여주는 흐름도이다.

도 2는 본 발명의 일 실시예에 따른 U-Net 아키텍처와 각 계층의 동작을 보여주는 도면이다.

그림 3은 오츠 알고리즘(Otsu algorithm)에 의한 산사태 균열 식별 방법을 보여주는 도면이다.

도 4는 레이블이 지정되고 예측된 결과에 따른 균열을 보여주는 도면이다.

도 5는 산사태 균열 크기를 계산하는 방법을 보여주는 도면이다.

도 6은 원본 이미지에 주석이 달린 마스크, 분할된 클래스 및 크랙 마스크 이미지 자체의 몇 가지 예를 보여주는 도면이다.

도 7은 학습 과정 동안 평가 지표의 변환을 (a) 손실, (b) 정밀도, (c) 재현율, 및 (d) F1 점수 각각에 대해서 보여주는 도면이다.

도 8은 테스트 데이터 세트에서 산사태 균열 감지의 예를 정밀도(P), 재현율(R), 및 F1 점수(F1)와 함께 보여주는 도면이다.

도 9는 새로운 이미지에서 산사태 균열의 계산 및 측정된 길이와 너비가 표시된 균열 감지의 실시예를 보여주는 도면이다.

도 10은 본 발명의 일 실시예에 따른 이미지 딥러닝 기반 산사태 조기 감지 장치의 개념도이다.

도 11은 본 발명의 일 실시예에 따른 이미지 딥러닝 기반 산사태 조기 감지 장치의 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0034] 이하, 본 발명을 실시하기 위한 구체적인 내용을 본 발명의 바람직한 실시예들을 기준으로 첨부 도면을 참조하여 상세히 설명한다. 이때, 하나의 실시예의 도면에 개시된 것으로, 다른 실시예의 도면에 개시된 구성 요소와 동일한 구성 요소에 대해서는 동일한 참조 번호를 부여하고, 다른 실시예에서의 설명이 동일하게 적용될 수 있으며, 이에 대한 자세한 설명은 생략할 수 있다. 또한, 본 발명과 관련된 공지 기능 혹은 구성은 공지 기술을 참조하고, 여기서는 그에 대한 상세한 설명은 간략히 하거나 생략한다.

[0035] 아울러 본 명세서에서 사용되는 용어들은 본 발명에서의 기능을 고려하면서 가능한 현재 널리 사용되는 일반적인 용어(general term)들이 사용되었으나, 이는 당 분야에 종사하는 기술자의 의도 또는 판례, 새로운 기술의 출현 등에 따라 달라질 수 있다. 또한, 특정한 경우는 발명자가 임의로 선정한 용어도 있으며 이 경우 해당되는 발명의 설명 부분에서 상세히 그 의미를 기재할 것이다. 따라서 본 명세서에서 사용되는 용어는 단순한 용어의 명칭이 아닌 그 용어가 가지는 의미와 본 발명의 전반에 걸친 내용을 토대로 정의되어야 한다.

[0036] 본 명세서 전체에서 어떤 부분이 어떤 구성 요소를 '포함'한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있음을 의미한다. 또한, 본 명세서에서 사용되는 '부'라는 용어는 FPGA 또는 ASIC과 같은 하드웨어 구성만이 아닌 소프트웨어 구성도 의미한다. 그렇지만 '부'는 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. '부'는 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다. 따라서 일례로서 '부'는 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성요소들과 같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로 코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들 및 변수들을 포함한다. 구성요소들과 '부'들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 '부'들로 결합되거나 추가적인 구성요소들과 '부'들로 더 분리될 수 있다.

[0037] 본 발명에서는 딥 러닝과 이미지 처리 기술을 사용하여 산사태 균열 및/또는 토석류 흐름을 자동으로 식별하고 치수를 측정함으로써, 시간이 지남에 따른 산사태 진행상태를 모니터링할 수 있으며, 산사태 진행상태에 따른 단계별 조기 경보가 가능하게 할 수 있다.

[0038] 이를 위하여, 본 발명에서는 U-Net 완전 컨볼루션 네트워크 모델과 Otsu 방법에 기반한 이미지 처리 기술을 사용하여 카메라 뷰에서 지반 균열을 식별하고 길이와 너비를 계산할 수 있다. U-Net 모델은 현장 균열 이미지와 결합된 실험실 경사 테스트를 통해 얻은 이미지 데이터 세트(510개 이미지)에서 학습 및 테스트되었다. U-Net 모델로 식별된 산사태 균열은 평균 정밀도 0.89, 평균 재현율 0.91, F1 점수 0.9를 제공하여 Otsu 방법에 의한 이미지 처리(정밀도 0.71, 재현율 0.70, F1 점수 0.65)보다 우수한 성능을 보였다. 이때, U-Net 모델에 의한 산사태 균열 치수의 정확도는 84.2~98.8% 범위이다. 따라서, 본 발명의 U-Net 모델을 이용한 산사태 균열 감지 성능은 Otsu 방법에 기반한 이미지 처리 기술을 사용한 경우에 비하여 현저한 효과가 있음을 알 수 있다.

[0039] 도 1에는 본 발명의 일 실시예에 따른 딥러닝 기반 경사지 균열 감지 및 균열 크기 측정 방법을 개략적으로 보여주는 흐름도가 도시되어 있다.

[0040] 도면을 참조하면, 본 발명의 딥러닝 기반 경사지 균열 감지 방법은, 미리 학습용 이미지 데이터 세트에서 산사태 균열을 감지할 수 있는 U-Net 모델을 구축하고 학습하여 균열 감지 모델을 생성하고(Step 1), 생성된 균열 감지 모델을 이용하여 새로이 입력되는 입력 이미지에서 균열을 감지하여 균열이 감지된 균열 감지 이미지를 생성하고, 균영 감지 이미지에서 이미지 처리(IPT)에 의하여 균열의 길이 및 너비를 계산(Step 2)할 수 있다.

[0041] 본 발명의 일 실시예에 따른 딥러닝 기반 경사지 균열 감지 방법은, 이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계; 상기 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 및 상기 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계;를 포함할 수 있다.

[0042] 이때, 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성할 수 있다.

[0043] 또한, 균열 감지 이미지는, 입력 이미지를 1개 채널의 회색 이미지로 변환하고, 회색 이미지의 각각의 픽셀이 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 큰 경우 균열에 해당하는 값을 할당하고, 균열을 포함할 확률이 설정값(예를 들어, 0.5) 보다 작거나 같은 경우 배경에 해당하는 값을 할당하여 생성될 수 있다.

[0044] 산사태 균열 감지 단계는 컴퓨터 비전의 핵심 요소인 의미론적 분할 단계를 포함할 수 있다. 산사태 균열 감지

단계는 입력 이미지의 각 픽셀을 분류하는 것으로, 산사태 균열의 분할에는 픽셀 수준 감지를 위해 이미지의 각 각의 픽셀을 균열과 배경을 두 가지 범주로 나누는 것이 포함될 수 있다.

[0045] 이때, 산사태 균열 감지 단계는 완전 컨볼루션 네트워크 기반의 U-Net 아키텍처가 적용될 수 있다. U-Net 아키텍처에 의하면 CNN보다 적은 학습 이미지 데이터 세트에 의하여 더 정확한 결과를 얻을 수 있게 된다.

[0046] 표 1에는 본 발명의 일 실시예에 따른 U-Net 아키텍처와 각 계층의 동작이 표시되어 있다.

표 1

[0047]

No	Layers	Image size	Operation	Convolution kernel
IN	Input	3 × 256 × 256		-
CB1	L1	64 × 256 × 256	Conv1 + BN + ReLU	64
	L2	64 × 256 × 256	Conv2 + BN + ReLU	64
	L3	64 × 128 × 128	MaxPooling	-
CB2	L4	128 × 128 × 128	Conv3 + BN + ReLU	128
	L5	128 × 128 × 128	Conv4 + BN + ReLU	128
	L6	128 × 64 × 64	MaxPooling	-
CB3	L7	256 × 64 × 64	Conv5 + BN + ReLU	256
	L8	256 × 64 × 64	Conv6 + BN + ReLU	256
	L9	256 × 32 × 32	MaxPooling	-
CB4	L10	512 × 32 × 32	Conv7 + BN + ReLU	512
	L11	512 × 32 × 32	Conv8 + BN + ReLU	512
	L12	512 × 16 × 16	MaxPooling	-
CB5	L13	1024 × 16 × 16	Conv9 + BN + ReLU	1024
	L14	1024 × 16 × 16	Conv10 + BN + ReLU	1024
CB6	L15	512 × 32 × 32	ConvTrans-1	512
	L16	1024 × 32 × 32	Concatenate-1	-
	L17	512 × 32 × 32	Conv11 + BN + ReLU	512
	L18	512 × 32 × 32	Conv12 + BN + ReLU	512
CB7	L19	256 × 64 × 64	ConvTrans-2	256
	L20	512 × 64 × 64	Concatenate-2	-
	L21	256 × 64 × 64	Conv13 + BN + ReLU	256
	L22	256 × 64 × 64	Conv14 + BN + ReLU	256
CB8	L23	128× 128 × 128	ConvTrans-3	128
	L24	256 × 128 × 128	Concatenate-3	-
	L25	128 × 128 × 128	Conv15 + BN + ReLU	128
	L26	128 × 128 × 128	Conv16 + BN + ReLU	128
CB9	L27	64 × 256 × 256	ConvTrans-4	64
	L28	128 × 256 × 256	Concatenate-4	-
	L29	64 × 256 × 256	Conv17 + BN + ReLU	64
	L30	64 × 256 × 256	Conv18 + BN + ReLU	64
	L31	1 × 256 × 256	Conv19 + Softmax	1
OUT	Output	1 × 256 × 256		

[0048] 표 1에서, Conv#은 컨볼루션 레이어를 나타내고, BN#은 일괄 정규화 계층을 나타내고, ReLU는 ReLU 활성화 함수를 나타내고, MaxPooling은 Max pooling Layer를 나타내고, ConvTrans#은 디콘볼루션 레이어를 나타내고, Softmax는 Softmax 활성화 함수를 나타내고, Concatenate#는 두 개의 이미지가 채널 수준에서 하나로 연결됨을 의미한다.

[0049] 도 2에는 본 발명의 일 실시예에 따른 U-Net 아키텍처와 각 계층의 동작이 도시되어 있다.

[0050] 도면을 참조하면, 입력 이미지에서 U-Net 아키텍처에 의하여 산사태 균열을 감지하여 분할(segmentation)할 수 있다. U-Net 아키텍처는 입력 이미지(Input), 수축 경로(CB1-CB5), 확장 경로(CB5-CB9) 및 출력 이미지(Output)를 포함할 수 있다.

[0051] U-Net 모델은, 2개 이상의 컨볼류션 블록을 포함하여 다운 샘플링하는 수축 경로, 및 2개 이상의 컨볼류션 블록을 포함하여 업 샘플링하는 확장 경로를 포함할 수 있다. 컨볼류션 블록이 컨볼류션(Convolution), 배치 정규화(Batch Normalization, BN), 및 ReLU (Rectified Linear Unit) 연산을 포함할 수 있다. 이때, 수축 경로는 최대 풀링(Maxpooling) 작업을 포함하나, 확장 경로는 최대 풀링(Maxpooling) 작업을 포함하지 않을 수 있다.

[0052] U-Net 아키텍처에서는 크기가 3 × 256 × 256픽셀인 입력 이미지(즉, 3개의 채널과 256픽셀의 측면 길이가 있는 RGB 이미지)의 입력 이미지(Input)가 신경망에 입력될 수 있다. 수축 경로(CB1~CB5)는 2 이상의 컨볼류션 블록(Convolution block, CB)을 포함할 수 있으며, 일 실시예로서 연속되는 5개의 컨볼루션 블록(CB1~CB5)을 포함할 수 있다.

[0053] 컨볼류션 블록(Convolution block, CB)은 2개의 3 × 3 컨볼루션과 배치 정규화(batch normalization, BN)와 ReLU (Rectified Linear Unit) 연산, 및 풀링 작업을 포함할 수 있다.

[0054] 각각의 컨볼류션 블록(Convolution block, CB)은 2개의 3 × 3 컨볼루션(Conv1 ~ Conv10)을 포함할 수 있다. 각각의 컨볼루션(Conv1 ~ Conv10)은 배치 정규화(batch normalization, BN)와 ReLU (Rectified Linear Unit) 연산으로 이어질 수 있다. 다음으로, 공간 차원을 줄이기 위해 2 × 2 최대 풀링 작업이 적용될 수 있다. 기능 채널의 수는 각 다운샘플링 동안 두 배로 증가하지만 공간 차원은 반으로 줄어듭니다. 다만, 수축 경로의 마지막 컨볼루션 블록(CB5)은 이후 확장 경로로 이어지므로, 공간 차원을 줄일 필요가 없어서 풀링 작업이 필요 없을 수 있다.

[0055] 확장 경로(CB6~CB9)는 2 이상의 컨볼류션 블록(Convolution block, CB)을 포함할 수 있으며, 일 실시예로서 연속되는 4개의 컨볼루션 블록(CB1~CB5)을 포함할 수 있다.

[0056] 확장 경로의 모든 단계는 기능 맵의 업샘플링과 2 × 2 전치 컨볼루션을 포함할 수 있습니다. 이 경우, 기능 채널의 수를 절반으로 줄일 수 있다. 또한, 연결은 축소 경로와 2개의 3 × 3 컨볼루션에서 해당 기능 맵과 연결하는 데 사용될 수 있으며, 각각 뒤에 배치 정규화(batch normalization, BN)와 ReLU (Rectified Linear Unit) 연산이 올 수 있다. 최종 레이어에서 1 × 1 컨볼루션은 채널을 원하는 클래스 수에 매핑하는 데 사용될 수 있다. 처리 후 최종 출력 이미지(2진 이미지) 크기는 1 × 256 × 256픽셀이 될 수 있다. 각 픽셀은 픽셀이 이진 이미지의 산사태 균열 또는 배경을 가리킬 확률을 나타내는 2차원 벡터에 해당할 수 있다. 입력 픽셀 값이 0.5보다 크면 해당 출력 픽셀은 흰색(크랙)이 되고, 입력 픽셀 강도가 0.5 이하이면 출력 픽셀은 검은색(배경)이 될 수 있다.

[0057] 한편, 다른 실시예로서, 도 3에 도시된 바와 같이 오츠 알고리즘(Otsu algorithm)에 의하여 입력 이미지에서 균열을 감지 및 분할(segmentation)할 수 있다.

[0058] 그림 3에는 오츠 알고리즘(Otsu algorithm)에 의한 산사태 균열 식별 방법이 도시되어 있다.

[0059] 도면을 참조하면, 오츠 알고리즘(Otsu algorithm)에 의한 산사태 균열 식별 방법은, (a) 입력 이미지(Input image) 입력단계, (b) 회색 이미지(Gray image) 변환단계, (c) 이미지 히스토그램 생성단계, (d) 이미지 픽셀 각각을 분류하는 픽셀 분류 단계, 및 (e) 최종 출력 이미지 생성단계를 포함할 수 있다.

[0060] 입력 이미지 입력단계(a)에는 RGB 입력 이미지를 입력받는다. 회색 이미지 변환단계(b)에는 RGB 입력 이미지(a)를 256 × 256픽셀 크기의 입력 회색 이미지(b)로 변환한다. 이미지 히스토그램 생성단계(c)에는 오츠 방법(오츠 알고리즘)을 사용하여 강도 수준 확률(그레이 수준 히스토그램 생성) 및 임계값 k 값을 계산한다. 픽셀 분류단계(d)에는 각각의 픽셀의 픽셀값이 k보다 크면 크랙(흰색), 픽셀값이 k보다 작거나 같으면 배경(검정색)으로 픽셀을 분류합니다.

[0061] 최종 출력 이미지 생성단계에는 출력 이미지의 노이즈를 줄이기 위한 중앙값 필터링을 수행하여, 256 × 256 픽셀 크기의 최종 흑백 이미지 출력할 수 있다. 필터링 알고리즘은 3 × 3 행렬을 사용하여 전체 출력 이미지를 스캔하고 행렬 내부의 모든 값의 중앙값을 간단히 고려하여 중심 픽셀의 값을 다시 계산할 수 있다.

[0062] 오츠 방법에 의하여 픽셀 강도를 기반으로 산사태 균열을 식별할 수 있다. 이때, 입력 픽셀 강도가 임계값(k)보다 큰 경우, 해당 출력 픽셀은 흰색(크랙)이 된다. 입력 픽셀 강도가 임계값보다 작거나 같으면 출력 픽셀은 검은색(배경)이 된다. 따라서, 임계값 k에 의해 모든 픽셀들은 두 개의 클래스 C1 또는 C2(배경 또는 균열)로 분리될 수 있다. C1은 [1,... ,k] 레벨의 픽셀을 나타내고, C2는 [k + 1,..., L] 레벨의 픽셀을 나타내며, L은 입력 회색 이미지의 최대 픽셀 값(255)이 되고, k 수준의 임계값은 클래스 간 분산(수학식 1)을 최대화하여 얻을 수 있다.

수학식 1

[0063] $$\sigma_B^2(k)=\omega_1(t)\omega_2(t)[\mu_1(t)-\mu_2(k)]^2$$

[0064] 최적 임계값 k*는 다음과 같습니다(수학식 2).

수학식 2

[0065] $$\sigma_B^2(k^*) = \max_{1 \le k \le L} \sigma_B^2(k)$$

[0066] 여기서 $\omega_1(k)$, $\omega_2(k)$는 두 클래스(C1, C2)의 확률이고(수학식 3, 4), $\mu_1(k)$, $\mu_2(k)$는 아래 식(수학식 5, 6)을 사용하여 각각 C1 및 C2에 대한 평균이다.

수학식 3

[0067] $$\omega_1(k) = \Pr(C1) = \sum_{i=1}^{k} Pi$$

수학식 4

[0068] $$\omega_2(k) = \Pr(C2) = \sum_{i=k+1}^{L} Pi = 1 - \omega_1(k)$$

수학식 5

[0069] $$\mu_1(k) = \sum_{i=1}^{k} iPr(i \mid C1) = \sum_{i=1}^{k} iPi/\omega_1(k)$$

수학식 6

[0070] $$\mu_2(k) = \sum_{i=k+1}^{L} iPr(i \mid C2) = \sum_{i=k+1}^{L} iPi/\omega_2(k)$$

[0071] 여기서 Pi는 i회색조가 존재할 확률이며, 이미지의 총 픽셀 수(N)에 대한 레벨 i의 픽셀 수(n_i)의 비율로 정의될 수 있다(수학식 7).

수학식 7

[0072] $$Pi = \frac{n_i}{N}$$

[0073] 도 4는 레이블이 지정되고 예측된 결과를 나타낸다.

[0074] 도면을 참조하면, 출력 이미지의 각각의 픽셀은 참양성(TP), 가양성(FP), 참음성(TN), 및 가음성(FN)으로 분할될 수 있다. 참양성(TP)은 크랙 픽셀이 올바르게 식별된 것이며, 가양성(FP)은 배경 픽셀이 균열 픽셀로 잘못 식별된 것이며, 참음성(TN)은 배경 픽셀이 올바르게 식별된 것이며, 가음성(FN)은 배경 픽셀로 식별된 균열 픽셀을 나타낸다.

[0075] 한편, 분할 알고리즘의 성능 평가는 정밀도(Precision), 재현율(Recall) 및 F1 점수에 의하여 이루어질 수 있다. 정밀도(Precision)는 모든 긍정적인 예측 중에서 올바른 긍정적인 예측의 비율을 나타내고, 재현율

(Recall)은 긍정적인 예측의 완전성을 고려한다(즉, 올바르게 분류된 균열 픽셀의 수). F1 점수는 정밀도와 재현율의 조화 평균에 해당한다. 정밀도, 재현율 및 F1 점수(F1)는 다음의 수학식 8, 9 및 10과 같다.

수학식 8

[0076] $$Precision = \frac{TP}{TP+FP}$$

수학식 9

[0077] $$Recall = \frac{TP}{TP+FN}$$

수학식 10

[0078] $$F1score = \frac{2 \times P \times R}{P+R}$$

[0079] 도 5에는 산사태 균열 크기를 계산하는 방법이 도시되어 있다. 도면을 참조하면, 본 발명의 실시예에서는 식별된 균열의 크기가 이미지 처리를 통하여 산출될 수 있다. 즉, 균열 크기 산출 단계에 균열 감지 이미지를 이미지 처리하여 식별된 균열 형상으로부터 균열의 길이와 너비를 계산할 수 있다.

[0080] 이때, 산사태 균열을 식별한 후 이미지 처리를 사용하여 균열 길이와 너비를 자동으로 계산할 수 있다. 균열 형상을 설정된 수평 또는 수직 픽셀 수의 단위로 복수의 세그먼트로 분할하고, 기준선과 균열의 경계의 두 교차점의 중심을 각각의 세그먼트의 중심점으로 하고, 해당 세그먼트의 중심점과 이전 세그먼트의 중심점 사이의 거리를 해당 세그먼트의 중심선으로 하고, 모든 세그먼트의 중심선의 합이 균열의 길이로 계산될 수 있다.

[0081] 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 0~45도이면 균열 감지 이미지에서 수직선이 상기 기준선이 되고, 균열의 시작점과 끝점을 잇는 선과 수평선 사이의 각도가 45~90도이면 균열 감지 이미지에서 수평선이 상기 기준선이 될 수 있다. 해당 균열 세그먼트에서 중심선에 수직인 선과 균열 경계의 교차하는 두 지점들 사이의 거리가 해당 균열 세그먼트의 너비가 될 수 있다. 전체 균열의 너비는 각각의 균열 세그먼트 너비들의 평균이 될 수 있다.

[0082] 즉, 도 5에서 (a) 및 (b)에 따르면, 균열 길이는 균열 중심선의 길이가 될 수 있으며, 균열 폭은 여러 균열 세그먼트 폭의 평균이 될 수 있다.

[0083] 이를 위하여 균열(crack)이 수평 균열(도 5의 (a) 및 (c))과 수직 균열(도 5의 (b) 및 (d))의 두 그룹으로 분할될 수 있다. 수평 균열은 각도가 0~45도인 균열이고, 수직 균열은 각도가 45~90도인 균열이 될 수 있다. 이때, 각도는 균열의 시작점(A)과 끝점(B)을 잇는 선과 수평선 사이의 각도이다. 작은 중심선 세그먼트의 길이를 계산할 때, 수직선(45~90도의 균열) 또는 수평선(0~45도의 균열) 각각 설정된 개수 예를 들어 5픽셀의 길이를 사용할 수 있다.

[0084] 한편, 수직선(수평 균열의 경우) 또는 수평선(수직 균열의 경우)과 균열 경계 사이의 두 교차점(x^u_i, y^u_i) 및 (x^l_i, y^l_i)이 결정될 수 있다. 이 경우, 두 교차점을 사용하여 균열 세그먼트의 중심점이 식별될 수 있다. 균열의 전체 길이(L)는 수학식 11의 유클리드 거리 공식을 사용하여 계산된 각 중심선 세그먼트의 길이의 합이 될 수 있다.

수학식 11

$$L=\sum_{i=1}^{n+1}\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2}$$

[0085]

[0086] 여기서, (x_i, y_i)와 (x_{i+1}, y_{i+1})는 두 개의 연속 균열 세그먼트의 중심점 좌표이고 n은 중심점의 총 개수이다. 균열 세그먼트 너비는 다음과 같이 계산할 수 있다. 각 중심점(x_i y_i)에서 각 중심선 균열 세그먼트에 수직인 선을 그리고 이 선이 균열 경계와 교차하는 두 점(x_t^i, y_t^i) 및 (x_i^b, y_i^b)를 감지할 수 있다. 이 지점들 사이의 거리는 균열 세그먼트의 너비가 될 수 있다. 전체 균열의 평균 너비($\overline{W}$)는 각각의 균열 세그먼트 너비들의 평균이 될 수 있다(수학식 12).

수학식 12

$$\overline{W}=\frac{\sum_{i=1}^{n}\sqrt{(x_i^t-x_i^b)^2+(y_i^t-y_i^b)^2}}{n}$$

[0087]

[0088] 픽셀 단위로 계산된 균열의 길이와 너비에 mm 단위 변환 계수를 곱하여 mm 단위의 균열의 길이와 너비를 계산할 수 있다. 이때, mm 단위 변환 계수는 mm 단위의 설정된 면적에 포함되는 가로 픽셀수 X 세로 픽셀수의 값으로 나눈 값의 제곱근이 될 수 있다.

[0089] 균열 크기의 픽셀 단위를 mm로 변환하기 위해 기록된 표면의 면적을 사용할 수 있다($A = a \times b$ in mm^2). 수학식 13, 14에 의하여 픽셀 단위가 mm 단위로 변환될 수 있다.

수학식 13

$$L(mm)=L(Pixel\ unit)\times\sqrt{\frac{A}{n\times m}}$$

[0090]

수학식 14

$$\overline{W}(mm)=\overline{W}(Pixel\ unit)\times\sqrt{\frac{A}{n\times m}}$$

[0091]

[0092] 여기서, *nxm*는 면적 *A = a x b* mm^2에 대응되는 픽셀들에 의하여 선택된 이미지의 너비와 높이를 곱한 것이다.

[0093] 표 2에는 균열 감지 및 균열 치수 계산을 위한 데이터 세트의 분할이 표시되어 있다.

표 2

Stage	Split dataset	U-Net	Otsu
Crack detection	Total images	510 (Original images)	-
	Training image (80%)	408 (Original images) 1224 (Variant images)	-
	Validation image (10%)	51 (Original images)	-
	Testing image (10%)	51 (Original images)	51 (Original images)
Calculation of crack dimension	10 (Original images)		

[0094]

[0095] U-Net 모델과 Otsu 방법을 사용하여 균열 감지를 위한 데이터 세트가 표 2와 같이 구축될 수 있다. 대부분의 균열 이미지는 녹색 잔디와 회색 땅을 포함할 수 있다. 필드 이미지는 다양한 거리와 해상도에서 촬영되었다. 모든 이미지는 이미지 분할 전에 256 × 256 픽셀로 크기가 조정될 수 있다. Labelme 도구가 510개 이미지에 산사태 균열 마스크를 만드는 데 사용될 수 있으며, 마스크는 별도로 레이블을 지정할 수 있도록 저장될 수 있다.

[0096] 도 6은 원본 이미지에 주석이 달린 마스크, 분할된 클래스 및 크랙 마스크 이미지 자체의 몇 가지 예를 보여준다. U-Net 학습 동안 데이터의 균형을 유지하기 위해 주석이 달린 이미지는 학습(80%, 408개 이미지), 검증(10%, 51개 이미지) 및 테스트(10%, 51개 이미지) 데이터 세트의 세 가지 데이터 세트로 무작위로 분할될 수 있다.

[0097] 408개의 이미지로 구성된 데이터 세트는 딥 러닝 모델을 적절하게 학습하기 위해서는 부족하므로, 데이터가 증강될 수 있다. 이를 위하여, 이미지를 수직으로 뒤집고 회전하고 밝기를 변경하였다. 추가 1,124개의 변형 이미지는 데이터 증강을 통하여 U-Net 모델의 데이터 세트 용으로 생성되었다.

[0098] 한편, Otsu 방법의 평가를 위한 데이터 세트는 이 방법이 학습 및 검증을 위한 데이터 세트를 필요로 하지 않기 때문에 테스트 데이터(51개 이미지)에서 파생되었다. 10개의 실험적 균열 이미지를 사용하여 균열 치수가 계산되었다. 표 2에는 균열 감지 및 균열 치수 계산을 위한 데이터 세트 분할이 보여진다.

[0099] 한편, U-Net 학습은 학습률은 0.0001, 배치 크기는 5, 학습은 40 에포크 이상 실행되었다. 많은 수의 이미지, 다양한 이미지 유형 및 명백한 균열 이미지는 모델의 학습 시간을 단축하고 나중에 균열 감지의 정확도를 높일 수 있다. 따라서, 3가지 증강 기법(이미지 뒤집기, 이미지 회전, 밝기 변경)에 의하여, 학습 이미지 수를 408개의 원본 이미지에서 1632개의 학습 이미지로 늘렸다. 학습 중 U-Net 모델의 평균 손실, 정밀도, 재현율 및 F1 값은 각 Epoch 이후에 업데이트되었다.

[0100] 도 7은 학습 과정 동안 평가 지표의 변환을 (a) 손실, (b) 정밀도, (c) 재현율, 및 (d) F1 점수 각각에 대해서 보여준다.

[0101] 손실은 학습 및 검증 세트에 대해 각각 Epoch 32에서 0.087 및 0.099에 도달했다(도 7a). U-Net 모델은 epoch 32 이후 검증 데이터 세트에서 0.93의 정밀도(도 7b), 0.89의 재현율(도 7c) 및 0.91의 F1 점수(도 7d)를 달성했다. 다만, 학습이 epoch 40까지 계속되었지만, 검증 세트 수치가 더이상 개선되지 않는다. 따라서, 우리는 epoch 32로 학습된 모델이 바람직할 수 있다. 이때, 가장 좋은 가중치가 저장될 수 있다. 테스트에서는 산사태 균열의 마스킹된 이미지 51개를 사용되었다.

[0102] 도 8에는 테스트 데이터 세트에서 산사태 균열 감지의 예가 정밀도(P), 재현율(R), 및 F1 점수(F1)와 함께 도시되어 있다. 도 8은 U-Net 모델과 Otsu 방법을 사용한 균열 식별의 몇 가지 예를 제시하며, 실험실 경사 실험에 의한 균열과 실제 균열이 포함된다.

[0103] 표 3은 테스팅 데이터셋에 대한 U-Net 모델과 Otsu 방법의 성능을 보여준다.

표 3

Method	Value	Precision	Recall	F1-score
U-Net	Min	0.55	0.74	0.67
	Max	0.99	0.97	0.97
	Average	**0.89**	**0.91**	**0.90**
Otsu	Min	0.04	0.47	0.08
	Max	0.97	1.00	0.93
	Average	**0.71**	**0.70**	**0.65**

[0104]

[0105] U-Net 모델은 평균 정밀도 0.89, 재현율 0.91, F1 0.90을 보여주며, Otsu 방법에 대한 해당 수치는 0.71, 0.70 및 0.65가 된다. 따라서, 본 발명의 일 실시예에 따른 U-Net 모델에 의한 균열 감지 성능이 정밀도, 재현율, F1 점수 각각의 측면에서 현저한 효과를 보여준다.

[0106] 도 8은 U-Net 모델과 Otsu 방법을 사용한 균열 식별의 몇 가지 예를 제시하며, 실험실 경사 실험에 의한 균열과 실제 균열이 포함된다. 도 8d는 Otsu 방법 오류의 몇 가지 예를 보여준다. Otsu 방법은 중앙값 알고리즘을 사용하여 노이즈를 제거했지만 균열 감지 정확도가 좋지 않았습니다. 여러 배경 픽셀 영역이 균열로 오인되었다(도 8d).

[0107] 하지만, U-Net을 사용한 경우에는 모든 이미지에서 높고 균일한 균열 감지 정확도를 제공했습니다. 대조적으로, Otsu 방법의 사용은 일관되지 않은 결과를 산출했다(높은 정확도와 낮은 정확도가 혼합됨). 정밀도 범위는 U-Net 및 Otsu 방법에 대해 각각 0.55~0.99 및 0.04~0.97이며, 재현율 범위는 각각 0.74~0.97 및 0.47~1이며, F1 범위는 각각 0.67~0.97 및 0.08~0.93이다.

[0108] 표 4 및 5 에는 각각 U-Net 모델과 Otsu 방법의 TP, FP, TN, FN, Precision, Recall, F1 값이 나열되어 있다. U-Net 모델은 배경이 다른 여러 유형의 산사태 균열과 증강 알고리즘에 의해 생성된 변형 이미지에 대해 학습되었다. 이러한 다양성을 감안할 때 모델은 높고 균일한 정확도(모든 테스트 이미지)를 제공한 것을 알 수 있다.

표 4

[0109]

No	Image	TP	FP	FN	TN	Precision	Recall	F1
1	Crack_08	1974	541	270	62751	0.78	0.88	0.83
2	Crack_24	2841	323	619	61753	0.90	0.82	0.86
3	Crack_39	3809	274	794	60659	0.93	0.83	0.88
4	Crack_42	12379	1048	668	51441	0.92	0.95	0.93
5	Crack_56	18183	1171	1212	44970	0.94	0.94	0.94
6	Crack_60	10491	302	675	54068	0.97	0.94	0.95
7	Crack_88	3398	390	470	61278	0.90	0.88	0.89
8	Crack_91	13008	654	1656	50218	0.95	0.89	0.92
9	Crack_101	8741	1301	975	54519	0.87	0.90	0.88
10	Crack_124	4578	900	233	59825	0.84	0.95	0.89
11	Crack_144	6308	488	776	57964	0.93	0.89	0.91
12	Crack_159	4675	1170	316	59375	0.80	0.94	0.86
13	Crack_172	4792	367	686	59691	0.93	0.87	0.90
14	Crack_185	1108	891	175	63362	0.55	0.86	0.67
15	Crack_190	1825	361	384	62966	0.83	0.83	0.83
16	Crack_200	1900	626	97	62913	0.75	0.95	0.84
17	Crack_225	2560	531	302	62143	0.83	0.89	0.86
18	Crack_236	10868	1419	750	52499	0.88	0.94	0.91
19	Crack_247	9658	702	1308	53868	0.93	0.88	0.90
20	Crack_270	7036	625	270	57605	0.92	0.96	0.94
21	Crack_286	14715	354	543	49924	0.98	0.96	0.97
22	Crack_288	3657	213	584	61082	0.94	0.86	0.90
23	Crack_299	5631	336	560	58809	0.95	0.91	0.93
24	Crack_303	9325	322	607	55282	0.97	0.94	0.95
25	Crack_306	24066	155	8420	32895	0.99	0.74	0.85
26	Crack_319	13688	979	705	50161	0.93	0.95	0.94
27	Crack_328	4985	544	667	59340	0.90	0.88	0.89
28	Crack_336	2197	816	157	62366	0.73	0.93	0.82
29	Crack_347	8320	2077	249	54890	0.80	0.97	0.88
30	Crack_350	5834	1630	200	57872	0.78	0.97	0.86
31	Crack_353	4388	533	596	60019	0.89	0.88	0.88
32	Crack_373	11453	411	1464	52208	0.97	0.89	0.93
33	Crack_376	20384	473	622	44057	0.98	0.97	0.97
34	Crack_384	6730	931	314	57561	0.88	0.96	0.92
35	Crack_390	7520	621	432	56963	0.92	0.95	0.93
36	Crack_404	3996	701	263	60576	0.85	0.94	0.89
37	Crack_427	5987	623	549	58377	0.91	0.92	0.91
38	Crack_438	5473	488	331	59244	0.92	0.94	0.93
39	Crack_454	4701	543	457	59835	0.90	0.91	0.90
40	Crack_457	5405	450	266	59415	0.92	0.95	0.93
41	Crack_470	6213	791	620	57912	0.89	0.91	0.90
42	Crack_486	4564	614	294	60064	0.88	0.94	0.91
43	Crack_488	2997	459	234	61846	0.87	0.93	0.90
44	Crack_492	4612	573	485	59866	0.89	0.90	0.89
45	Crack_494	8009	415	557	56555	0.95	0.93	0.94
46	Crack_495	8992	435	678	55431	0.95	0.93	0.94
47	Crack_498	6724	479	764	57569	0.93	0.90	0.91
48	Crack_502	18996	318	3358	42864	0.98	0.85	0.91
49	Crack_508	4511	501	272	60252	0.90	0.94	0.92
50	Crack_514	7524	542	219	57251	0.93	0.97	0.95
51	Crack_517	8570	581	356	56029	0.94	0.96	0.95
Average		0.89		0.91		0.9		

[0110] 대조적으로, Otsu 자동 이미지 임계값은 노이즈의 영향을 받는다. 바이모달 히스토그램의 피크 사이의 날카로운 계곡이 저하되고 이 방법은 실패할 수 있다. 또한 낮거나 불균일한 조명에서 Otsu 방법으로 균열을 이미지화하면 히스토그램이 더이상 바이모달이 아니므로 분할이 실패될 수 있다.

표 5

No	Image	TP	FP	FN	TN	Precision	Recall	F1
1	Crack_08	1253	2972	991	60320	0.30	0.56	0.39
2	Crack_24	2050	9945	1410	52131	0.17	0.59	0.26
3	Crack_39	2426	2227	2177	58706	0.52	0.53	0.52
4	Crack_42	8411	2598	4636	49891	0.76	0.64	0.69
5	Crack_56	11141	4250	8254	41891	0.72	0.57	0.64
6	Crack_60	9787	1219	1379	53151	0.89	0.88	0.88
7	Crack_88	3853	30076	15	31592	0.11	1.00	0.20
8	Crack_91	7134	6581	7530	44291	0.52	0.49	0.50
9	Crack_101	6107	2052	3609	53768	0.75	0.63	0.68
10	Crack_124	3713	7174	1098	53851	0.34	0.77	0.47
11	Crack_144	3855	256	3229	58196	0.94	0.54	0.69
12	Crack_159	3472	1074	1519	59471	0.76	0.70	0.73
13	Crack_172	3553	2069	1925	57989	0.63	0.65	0.64
14	Crack_185	659	1072	624	63181	0.38	0.51	0.44
15	Crack_190	2089	54190	120	9137	0.04	0.95	0.08
16	Crack_200	1986	53330	11	10209	0.04	0.99	0.08
17	Crack_225	2407	5502	455	57172	0.30	0.84	0.44
18	Crack_236	8594	916	3024	53002	0.90	0.74	0.81
19	Crack_247	8873	886	2093	53684	0.91	0.81	0.86
20	Crack_270	6221	1038	1085	57192	0.86	0.85	0.85
21	Crack_286	11849	806	3409	49472	0.94	0.78	0.85
22	Crack_288	4096	38579	145	22716	0.10	0.97	0.18
23	Crack_299	5461	2125	930	57020	0.72	0.85	0.78
24	Crack_303	7646	376	2286	55228	0.95	0.77	0.85
25	Crack_306	23664	817	8822	32233	0.97	0.73	0.83
26	Crack_319	8173	1309	6223	49631	0.86	0.57	0.69
27	Crack_328	2882	218	2770	59666	0.93	0.51	0.66
28	Crack_336	1108	324	1246	62858	0.77	0.47	0.58
29	Crack_347	7545	3443	1024	53824	0.69	0.88	0.77
30	Crack_350	5233	1252	801	58250	0.81	0.87	0.84
31	Crack_353	3632	8729	1352	51823	0.29	0.73	0.42
32	Crack_373	11246	13731	1671	38888	0.45	0.87	0.59
33	Crack_376	19217	1053	1789	43477	0.95	0.91	0.93
34	Crack_384	4204	308	2840	58184	0.93	0.60	0.73
35	Crack_390	4050	220	3902	57364	0.95	0.51	0.66
36	Crack_404	2197	317	2062	60960	0.87	0.52	0.65
37	Crack_427	4232	242	2304	58758	0.95	0.65	0.77
38	Crack_438	3998	236	1806	59496	0.94	0.69	0.80
39	Crack_454	3402	312	1756	60066	0.92	0.66	0.77
40	Crack_457	4232	223	1439	59642	0.95	0.75	0.84
41	Crack_470	4427	405	2406	58298	0.92	0.65	0.76
42	Crack_486	2460	259	2398	60419	0.90	0.51	0.65
43	Crack_488	2465	261	766	62044	0.90	0.76	0.82
44	Crack_492	2862	295	2236	60144	0.91	0.56	0.69
45	Crack_494	5241	5304	3325	51666	0.50	0.61	0.55
46	Crack_495	7337	1968	2333	53898	0.79	0.76	0.77
47	Crack_498	5760	541	1728	57507	0.91	0.77	0.83
48	Crack_502	12011	3489	10343	39693	0.77	0.54	0.63
49	Crack_508	3486	253	1297	60500	0.93	0.73	0.82
50	Crack_514	5694	265	2049	57528	0.96	0.74	0.84
51	Crack_517	6095	302	2831	56308	0.95	0.68	0.79
Average						**0.71**	**0.7**	**0.65**

[0111]

[0112] 표 6에는 본 발명에 따라 U-Net을 사용한 경우와 비교예(SVM, RetinaNet)와 균열 감지 성능을 비교한 결과를 보여준다.

[0113] 서포트 벡터 머신(SVM) 알고리즘을 사용하여 이미지를 크랙 유무에 따라 분류하는 경우, 학습을 위해 UAV 원본 이미지에서 RGB 이미지(50 × 50픽셀)를 잘라내고, 이미지 임계값은 균열 이미지에서 균열을 추출하는 데 사용되었다. 그들의 결과는 균열 분류의 정확도가 89.5%임을 보여주었다. 하지만, 이 방법은 이미지에서 크랙 유무를 분류할 수 있지만, 자동 균열 감지 기능이 없으며 이미지의 균열 위치를 결정할 수 없다.

표 6

Method of crack detection	Algorithm description	Image type	Performance	
			Accuracy of crack detection	Error of crack calculation
Support vector machine (SVM)	SVM algorithm classifies only no crack images and crack images. It cannot detect cracks on image automatically.	UAV	89.5 %	≤ 10% (crack length) ≤ 19% (crack width)
RetinaNet	RetinaNet model can build a bounding box corresponding to each crack in the image. But it cannot detect the shape of the crack in the box.	UAV and satellite	0.733 (AUC)	The study did not calculate the crack dimensions
U-Net	U-Net model can detect automatically crack and determine their shape.	RGB	90 % (F1 score)	≤ 6 % (crack length) ≤ 15.8% (crack width)

[0114]

[0115] RetinaNet을 사용하는 비교예는 UAV 이미지와 과거 위성 이미지에서 물체 감지 알고리즘을 사용하여 자동 산사태 균열을 감지한다. RetinaNet 모델은 0.733의 곡선 아래 면적(AUV) 값을 달성했다. 하지만, RetinaNet은 이미지의 균열 수를 감지하고, 각 균열에 해당하는 경계 상자를 만들 수 있다. 하지만, RetinaNet은 균열 치수 계산을 위한 경계 상자에서 균열의 모양을 식별할 수 없다.

표 7

[0116]

Crack	Cal. Length (mm)	Meas. Length (mm)	Accuracy (%)	Relative Error (%)	Cal. average width (mm)	Meas. average width (mm)	Accuracy (%)	Relative Error (%)
Crack_01	281	299	94.0	6.0	17	16	93.7	6.3
Crack_02	322	336	95.8	4.2	17	20	85.0	15.0
Crack_03	271	275	98.5	1.5	16	19	84.2	15.8
Crack_04	351	360	97.4	2.6	18	21	85.7	14.3
Crack_05	362	366	98.8	1.2	14	16	87.5	12.5
Crack_06	311	328	94.8	5.2	17	19	89.5	10.5
Crack_07	220	232	94.8	5.2	15	17	88.2	11.8
Crack_08	332	338	98.2	1.8	22	24	91.7	8.3
Crack_09	351	356	98.6	1.4	17	20	85.0	15.0
Crack_10	320	330	97.0	3.0	23	26	88.5	11.5

[0117] 하지만, 제안된 U - Net 모델은 90%의 정확도(F1 점수)로 이미지의 모든 픽셀에 레이블(균열 또는 배경)을 할당하여 균열을 자동으로 감지할 수 있다. 따라서 이미지 임계값을 사용하지 않고 균열의 모양을 직접 결정할 수 있다. 따라서, 본 발명의 일 실시에에 따른 균열 감지 방법은 균열 감지의 정확도를 높이고 균열 크기를 계산하는 시간을 줄일 수 있다.

[0118] 도 9에는 본 발명의 일 실시예에 따른 균열 감지 방법에 의한 균열 감지의 실시예가 도시되어 있다. 여기에는 새로운 이미지에서 산사태 균열의 계산 및 측정된 길이와 너비가 표시되어 있다.

[0119] 도면을 참조하면, U-Net 모델이 균열을 찾은 후 10개의 시험 이미지에서 균열 길이와 너비를 계산하기 위해 이미지 처리 방법이 사용되었다. 10개의 이미지에 대한 산사태 균열의 길이와 너비를 측정한 결과는 AutoCAD의 기능을 이용하여 수행하였다. 표 7에 계산 및 측정된 균열 치수와 오차가 나열되어 있다. 계산된 길이 오차의 범위는 1.2~6.0%로 94.0~98.8%의 정확도에 해당한다. 계산된 너비 오차 범위는 6.3~15.8%이며 정확도는 84.2~93.7%이다. 전체 오차 범위는 1.2~15.8%로 84.2~98.8%의 정확도에 해당한다. 따라서, 이러한 정확한 방법은 시간이 지남에 따라 산사태의 발달을 모니터링하여 조기 경보를 용이하게 할 수 있습니다.

[0120] 표면 균열의 발달과 전파는 미래 산사태의 중요한 지표가 될 수 있다.

[0121] 산사태는 인장 또는 전단 응력, 또는 경사 덩어리의 측면 팽창에 의해 유발될 수 있는데, 산사태가 시작되면 슬라이딩 바디 와 정지 바디 사이에 지반 균열이 형성되고 미끄럼 몸체 내에서 다른 표면 균열도 관찰될 수 있다. 슬라이딩 바디가 지속적으로 변형되면 균열 치수가 시간이 지남에 따라 증가한다.

[0122] 산사태 균열의 길이와 너비를 모니터링하면 이동 궤적이 빠르게 나타날 수 있다. 종래에는 신율계, 경사계 또는 경사계가 슬라이딩 방향을 따라 다양한 지점에서 표면 변위 또는 수직 이동을 측정하는 데 사용될 수 있다. 이러한 도구는 매우 신뢰할 수 있지만, 정확성이 보장되어야 하는 경우 높은 설치, 운영 및 유지보수 비용, 제한된 공간 범위를 갖는 연속적이지 않은 이산 관측, 현장에서 데이터 관리 및 처리의 어려움 등의 문제점이 있어서, 널리 사용되고 있지는 못하다.

[0123] 한편, 모바일 장치의 발전으로 인해 산사태를 모니터링하기 위해 고해상도 디지털 카메라가 점점 더 많이 사용되고 있다. 표면 균열 발달의 자동 기록은 산사태 균열을 유용하게 모니터링할 수 있다. 카메라는 불안정한 경사를 모니터링하는 데 사용될 수 있다.

[0124] 또한, 고해상도 항공 이미지에서 균열 후보를 추출하기 위해 가우시안 정합 필터링 알고리즘을 사용하는 이미지 처리가 사용될 수 있다. 이 경우, 모든 영향을 받는 지역과 균열 밀도 및 방향의 여러 지도에서 참양성률은 최대 65%, 위양성률은 10% 미만이다.

[0125] 또한, 산사태 균열을 식별하고 모션 전후의 방위각과 변위를 계산하기 위해 이미지 처리 방법(Otsu 알고리즘 및 Canny edge 감지)이 사용될 수 있다. 이러한 방법은 간단하고 사전 학습 데이터가 많이 필요하지 않다. 하지만, 그 정확도는 Otsu 세분화 임계값에 따라 달라지게 되며, Otsu 임계값은 픽셀을 개체 또는 배경으로 분류할 때만 잘 작동될 수 있다. 표면 배경에 색상이 다른 많은 객체가 포함된 경우 Otsu 임계값에 따른 픽셀 분류는 높은 정확도를 얻지 못할 수 있다.

[0126] 디지털 이미지 또는 비디오를 사용하는 산사태 모니터링을 위해 많은 딥 러닝 기반 컴퓨터 비전 접근 방식이 사용될 수 있다. CNN(Convolutional Neural Network)을 사용하여 공개적으로 사용 가능한 위성 이미지에서 산사태를 식별할 수도 있다. 심층 CNN과 이미지 처리 프레임워크를 사용하여 토석류의 움직임을 식별하고 전방 속도를 추정할 수 있다.

[0127] 한편, 딥 러닝 기반 컴퓨터 비전 알고리즘을 사용하여 콘크리트, 석조 벽 및 포장과 같은 구조 균열을 감지할 수 있다. 지상 균열 모니터링에서 위성 원격 감지 영상과 무인 항공기(UAV)를 사용하여 지상 균열을 조사할 수도 있다. 또한, 머신 러닝 분류 알고리즘을 적용하여 이미지를 크랙의 유무에 따라 분류한 다음 이미지 처리를 사용하여 크랙 치수를 계산할 수 있다. 이 경우, 이미지를 크랙의 유무에 따라 분류할 수 있으나 시간 경과에 따른 균열 발달의 자동 감지 및 치수 모니터링이 어려울 수 있다.

[0128] 한편, UAV 이미지에서 산사태 균열을 감지하기 위해 Retina-Net 이라는 물체 감지 알고리즘을 사용하여 자동으로 산사태 균열을 감지할 수 있다. RetinaNet 모델은 0.733의 곡선 아래 면적(AUV) 값을 달성했다. 하지만, RetinaNet은 이미지의 균열 수를 감지하고 각 균열에 해당하는 경계 상자를 만들 수 있으나, 균열 치수 계산을 위한 경계 상자에서 균열의 모양을 식별할 수 없다.

[0129] 분류, 물체 감지 및 이미지 분할은 컴퓨터 비전에서 널리 사용되는 중요한 알고리즘이다. 이미지 분할 알고리즘은 이미지의 객체들을 다수의 영역으로 분리할 수 있다. 또한 이미지의 각 개체에 대해 픽셀 단위 마스크를 생성할 수 있다. 이 기술을 사용하면 이미지에 있는 물체의 모양과 크기를 더 자세히 이해할 수 있게 된다.

[0130] U-Net 완전 컨볼루션 네트워크는 의미론적 분할 작업에서 핵심 알고리즘을 이용할 수 있다. 또한, 디지털 카메라에서 RGB 이미지를 Red-Green-Blue 채널에서 얻을 수 있다. 이때, 디지털 이미지 기록 장치의 관점에서 산사태 균열을 감지하고 지역화하기 위하여 U-Net 완전 컨볼루션 네트워크를 사용할 수 있다. 본 발명에서는 고정된 카메라를 통해 촬영된 영상을 이용하여 균열 크기(길이와 너비)를 계산하기 위하여 U-Net 모델과 결합된 새로운 영상처리 기반 프레임워크를 활용할 수 있다. 한편, U-Net 모델과 비교하기 위해 이미지 처리에 의한 분할 방법인 오츠(Otsu) 알고리즘이 사용될 수 있다.

[0131] 실험실 기울기 테스트를 통하여 학습용 데이터가 생성될 수 있다. 또한 U-Net 성능을 향상시키기 위해 실제 산사태 균열 데이터가 수집될 수 있다. 본 발명의 일 실시예에 따르면, 인공 지능과 이미지 처리의 새로운 결합 기술의 도움으로 산사태 균열을 감지하고 크기를 계산하여 조기 경보 및 모니터링을 용이하게 할 수 있다.

[0132] 도 10에는 본 발명의 일 실시예에 따른 이미지 딥러닝 기반 산사태 조기 감지 장치의 개념도가 도시되어 있다. 도 11에는 본 발명의 일 실시예에 따른 이미지 딥러닝 기반 산사태 조기 감지 장치의 블록도가 도시되어 있다. 본 실시예에 따른 산사태 조기 감지 장치는 도 1 내지 9에 도시된 딥러닝 기반 경사지 균열 감지 방법의 기술적 특징들이 모두 적용될 수 있다.

[0133] 도면을 참조하면, 이미지 딥러닝 기반 산사태 조기 감지 장치는, 영상 입력부(10); 영상 분석부(21, 22); 산사태 위험 판단부(30); 및 경고부(40);를 포함할 수 있다.

[0134] 영상 입력부(10)는 이미지 데이터 세트의 입력 이미지를 입력받을 수 있다.

[0135] 영상 분석부(20)는 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하고(21), 균열 감지 이미지에서 균열의 크기를 계산할 수 있다(22). 산사태 위험 판단부(30)는 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단할 수 있다. 경고부(40)는 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성할 수 있다.

[0136] 이때. 산사태 균열의 감지는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 입력 이미지에서 균열 감지 이미지를 생성하여 이루어질 수 있다.

[0137] 또한, 제1 파라미터(Parameter 1)와 제2 파라미터(Parameter 2)의 크기 비율에 따라 산사태 발생 위험 여부를 판단할 수 있다. 이때, 제1 파라미터는 균열의 크기로부터 결정되는 값이 될 수 있으며, 제2 파라미터는 토석류의 움직임 정보로부터 결정되는 값이 될 수 있다.

[0138] 한편, 산사태 균열의 크기는 균열의 길이와 너비를 포함할 수 있다. 균열의 길이와 균열의 너비의 크기 비율에 따라 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계 중의 어느 하나로 판단할 수 있다.

[0139] 이때, 주의 단계, 경고 단계, 및 대피 단계는 도 12의 산사태 위험 판단부(30)에 도시된 그래프에서와 같이, 제1 파라미터와 상기 제2 파라미터의 관계가 동일한 크기의 음의 기울기를 서로 다른 직선들에 의하여 결정될 수

있다.

[0140] 토석류의 움직임 정보는 토석류의 속도 및 토석류의 크기에 대한 정보를 포함할 수 있다. 이때, 토석류의 움직임은 YOLO 모델의 딥러닝에 의하여 식별될 수 있다. 또한, 토석류의 속도 및 크기는 이미지 처리에 의하여 계산될 수 있다. 또한, 토석류의 크기는 시간당 부피로 계산될 수 있으며, 이미지 처리에 의하여 계산되는 토석류의 속도, 폭, 및 깊이 정보에 의하여 계산될 수 있다.

[0141] 다른 실시예로서, 토석류의 깊이 정보는 별도로 구비되는 라이다 등의 센서를 통하여 획득될 수 있다.

[0142] 한편, 본 발명의 일 실시예에 따른 이미지 딥러닝 기반 산사태 조기 감지 방법은 도 10 내지 11에 도시된 이미지 딥러닝 기반 산사태 조기 감지 장치의 기술적 특징들이 모두 적용될 수 있다.

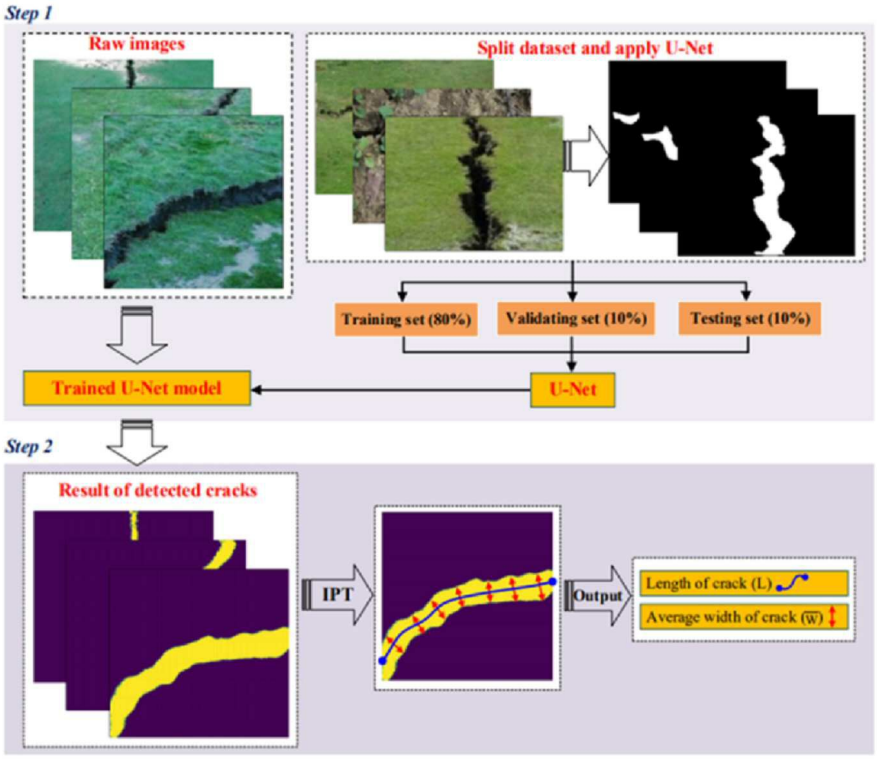
[0143] 이미지 딥러닝 기반 산사태 조기 감지 방법은, 이미지 데이터 세트의 입력 이미지를 입력받는 이미지 입력 단계; 입력 이미지에서 산사태 균열을 감지하여 균열이 배경에 대하여 구분하여 표시되는 균열 감지 이미지를 생성하는 산사태 균열 감지 단계; 균열 감지 이미지에서 균열의 크기를 계산하는 균열 크기 산출 단계; 산사태 균열의 크기로부터 산사태 발생 위험 여부를 판단하는 단계; 및 상기 산사태 발생 위험 여부에 따라 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 하나의 경고 신호를 생성하는 경고 단계를 포함할 수 있다.

[0144] 이때, 산사태 균열 감지 단계는 미리 학습용 이미지 데이터에 의하여 U-Net 모델을 이용하여 학습되어 생성되는 균열 감지 모델을 이용하여 상기 입력 이미지에서 균열 감지 이미지를 생성할 수 있다.

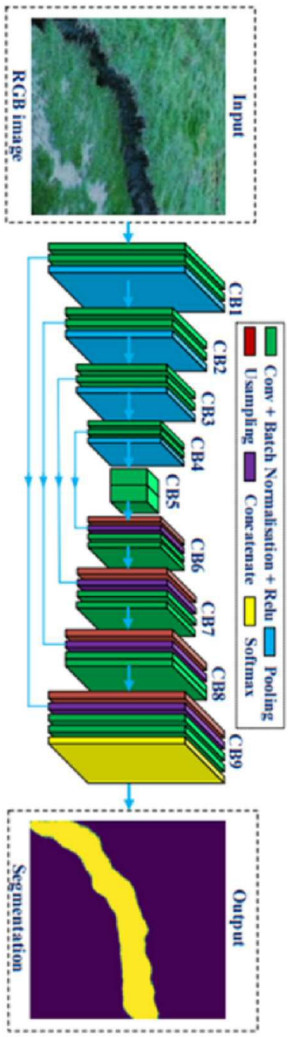
[0145] 이제까지 본 발명의 기술적 사상을 그 사상의 구체성을 담보하는 본 발명의 바람직한 실시예의 개시를 통해 개진하였다. 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 그 바람직한 실시예가 본 발명의 기술적 사상(본질적 특성)에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다. 그러므로 개시된 실시예는 한정적 관점이 아닌 설명적 관점에서 고려되어야 하며, 본 발명의 권리범위에는 청구범위에 개시된 사항뿐만 아니라 이와 균등한 범위 내에 있는 모든 차이도 포함되는 것으로 해석되어야 할 것이다. 특히, 하나의 실시예에서 설명된 기술적 특징은 다른 실시예에서도 적용될 수 있다.

도면

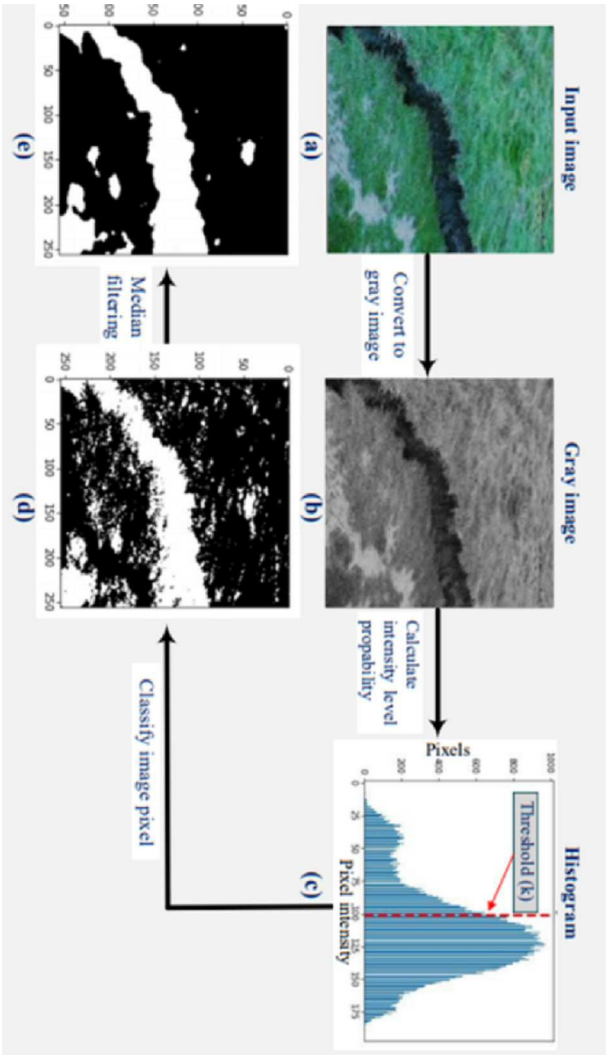
도면1



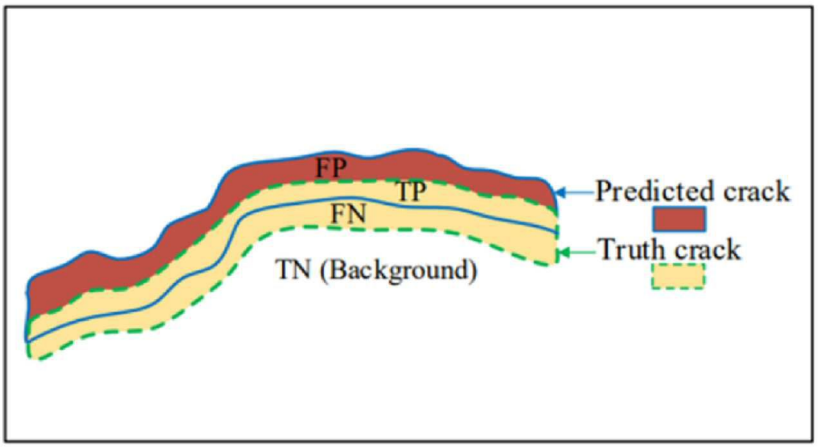
도면2


Input
RGB image
CB1
CB2
CB3
CB4
CB5
CB6
CB7
CB8
CB9
Conv + Batch Normalisation + Relu
Pooling
Upsampling
Concatenate
Softmax
Output
Segmentation

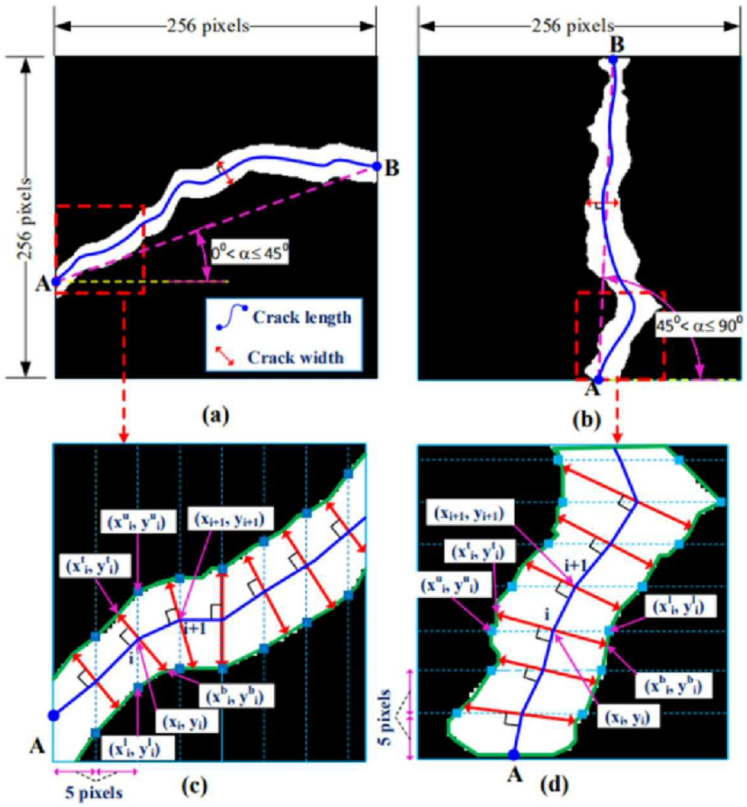
도면3


(a)
Input image
Convert to gray image
(b)
Gray image
Calculate intensity level propability
(c)
Histogram
Pixels
Threshold (k)
Pixel intensity
Classify image pixel
(d)
Median filtering
(e)

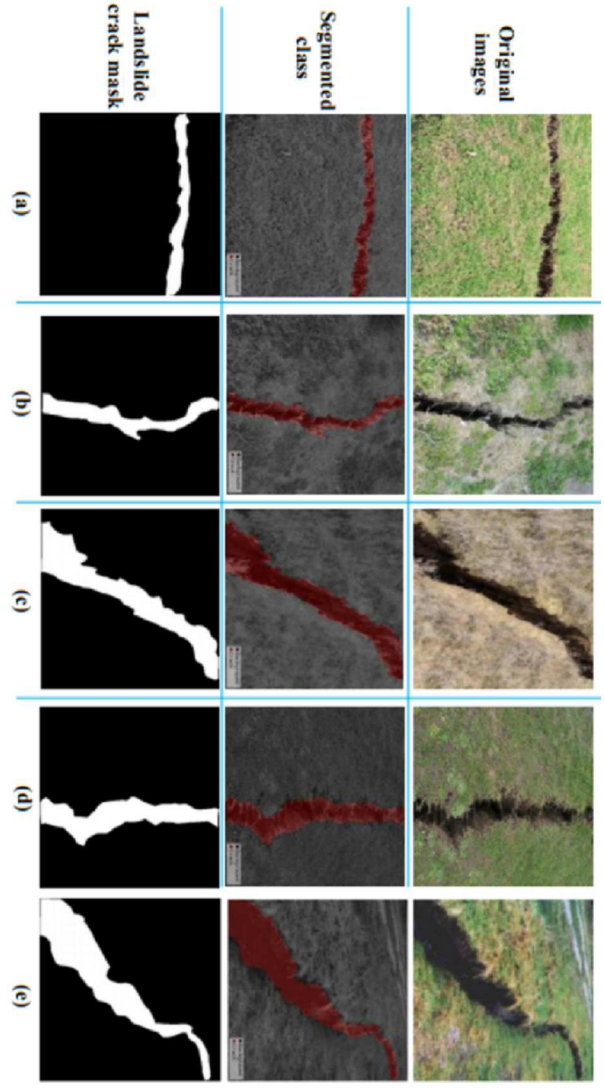
도면4


FP
TP
FN
TN (Background)
Predicted crack
Truth crack

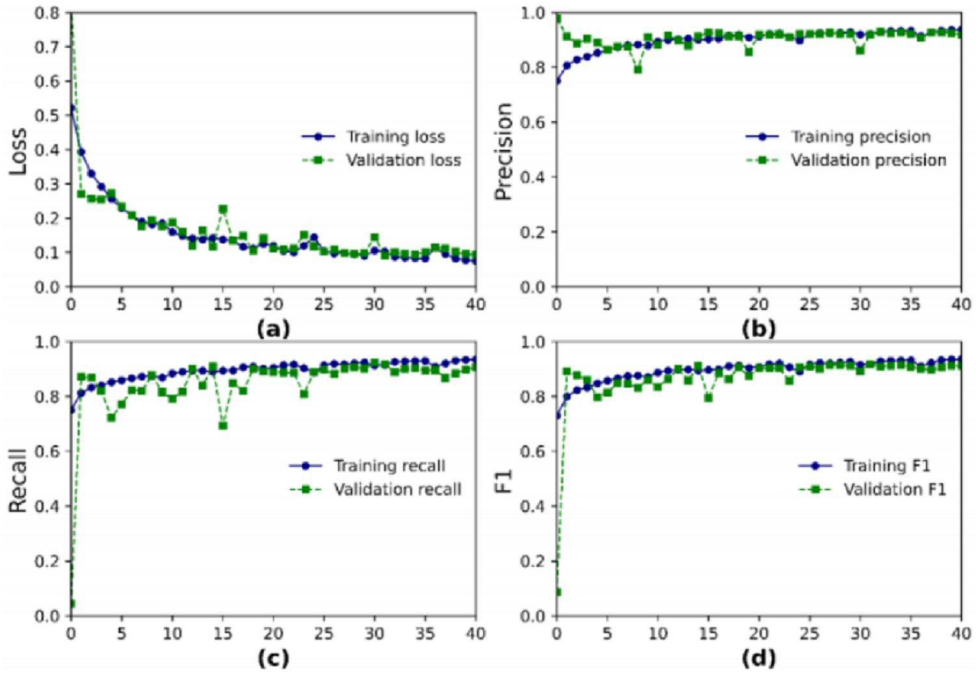
도면5


256 pixels
256 pixels
A
B
$0^0 < \alpha \leq 45^0$
Crack length
Crack width
(a)
256 pixels
B
A
$45^0 < \alpha \leq 90^0$
(b)
(x^u_i, y^u_i)
(x_{i+1}, y_{i+1})
(x^t_i, y^t_i)
i+1
i
(x^b_i, y^b_i)
(x_i, y_i)
(x^l_i, y^l_i)
A
5 pixels
(c)
(x_{i+1}, y_{i+1})
(x^t_i, y^t_i)
(x^u_i, y^u_i)
i+1
i
(x^l_i, y^l_i)
(x^b_i, y^b_i)
(x_i, y_i)
5 pixels
A
(d)

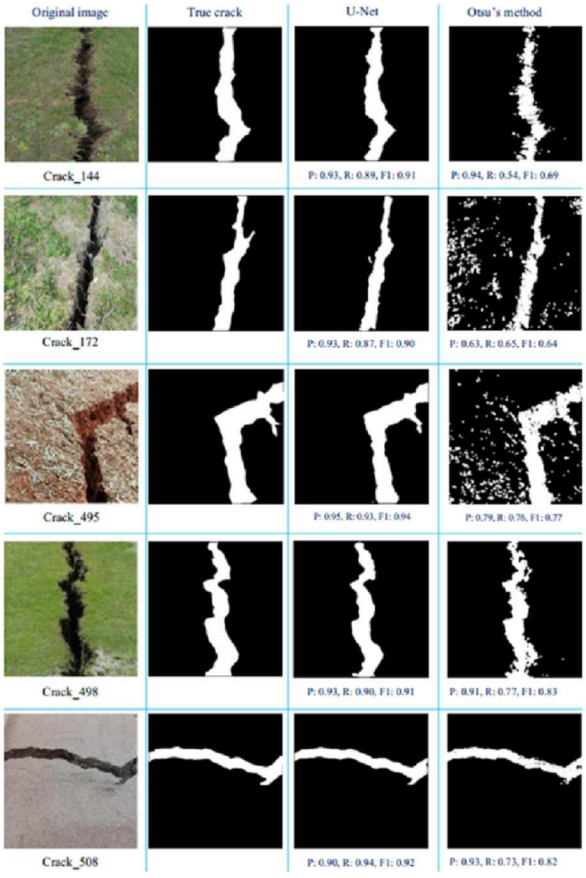
도면6


Original images
Segmented class
Landslide crack mask
(a)
(b)
(c)
(d)
(e)

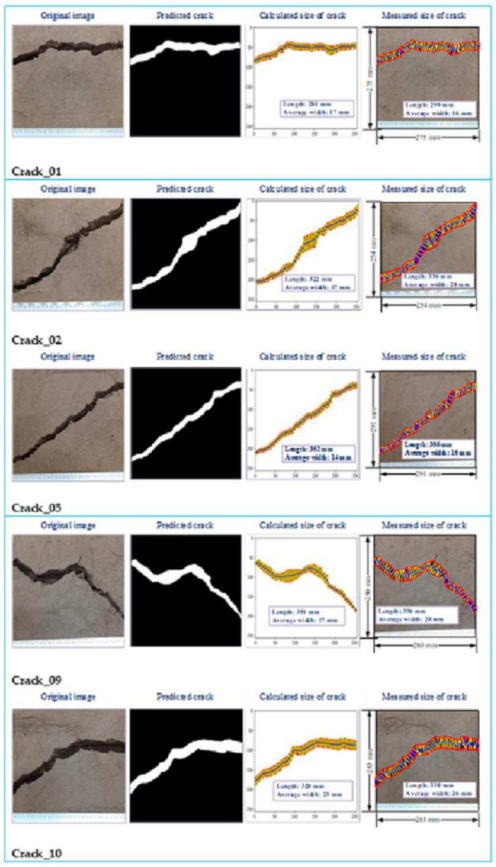
도면7


Loss
Training loss
Validation loss
(a)
Precision
Training precision
Validation precision
(b)
Recall
Training recall
Validation recall
(c)
F1
Training F1
Validation F1
(d)

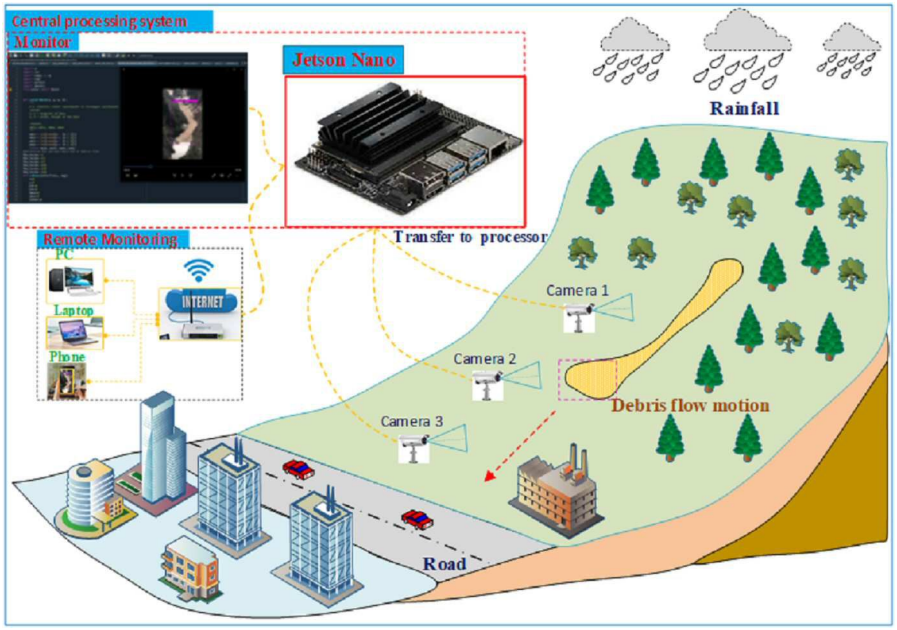
도면8


Original image
True crack
U-Net
Otsu's method
Crack_144
P: 0.93, R: 0.89, F1: 0.91
P: 0.94, R: 0.54, F1: 0.69
Crack_172
P: 0.93, R: 0.87, F1: 0.90
P: 0.63, R: 0.65, F1: 0.64
Crack_495
P: 0.95, R: 0.93, F1: 0.94
P: 0.79, R: 0.76, F1: 0.77
Crack_498
P: 0.93, R: 0.90, F1: 0.91
P: 0.91, R: 0.77, F1: 0.83
Crack_508
P: 0.90, R: 0.94, F1: 0.92
P: 0.93, R: 0.73, F1: 0.82

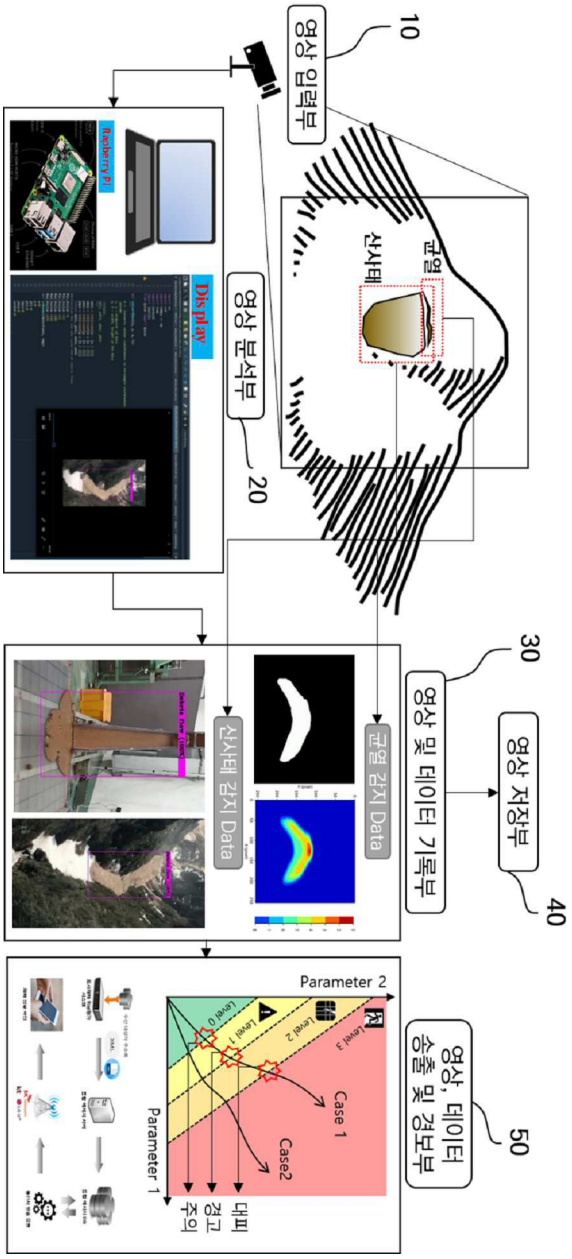
도면9


Original image
Predicted crack
Calculated size of crack
Measured size of crack
Crack_01
Crack_02
Crack_05
Crack_09
Crack_10

도면10


Central processing system
Monitor
Jetson Nano
Rainfall
Transfer to processor
Remote Monitoring
PC
Laptop
Phone
INTERNET
Camera 1
Camera 2
Camera 3
Debris flow motion
Road

도면11



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 12

【변경전】

제9항에 있어서,

제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 어느 하나로 판단하는 산사태 조기 감지 방법.

【변경후】

제10항에 있어서,

제1 파라미터와 제2 파라미터의 크기 비율에 따라 상기 산사태 발생 위험 여부를 주의 단계, 경고 단계, 및 대피 단계를 포함하는 위험 단계들 중의 어느 하나로 판단하는 산사태 조기 감지 방법.