

# Deep Learning Models for Autonomous Crack Detection System

Ji HongGeun<sup>†</sup> · Kim Jina<sup>††</sup> · Hwang Syjung<sup>†††</sup> · Kim Dogun<sup>††††</sup> · Park Eunil<sup>†††††</sup> ·  
Kim Young Seok<sup>††††††</sup> · Ryu Seung Ki<sup>†††††††</sup>

## ABSTRACT

Cracks affect the robustness of infrastructures such as buildings, bridge, pavement, and pipelines. This paper presents an automated crack detection system which detect cracks in diverse surfaces. We first constructed the combined crack dataset, consists of multiple crack datasets in diverse domains presented in prior studies. Then, state-of-the-art deep learning models in computer vision tasks including VGG, ResNet, WideResNet, ResNeXt, DenseNet, and EfficientNet, were used to validate the performance of crack detection. We divided the combined dataset into train (80%) and test set (20%) to evaluate the employed models. DenseNet121 showed the highest accuracy at 96.20% with relatively low number of parameters compared to other models. Based on the validation procedures of the advanced deep learning models in crack detection task, we shed light on the cost-effective automated crack detection system which can be applied to different surfaces and structures with low computing resources.

Keywords : Surface Inspection, Crack Detection, Computer Vision, Deep Learning

## 자동화 균열 탐지 시스템을 위한 딥러닝 모델에 관한 연구

지 흥근<sup>†</sup> · 김 지나<sup>††</sup> · 황 시정<sup>†††</sup> · 김 도건<sup>††††</sup> ·  
박은일<sup>†††††</sup> · 김영석<sup>††††††</sup> · 류승기<sup>†††††††</sup>

## 요약

균열은 건물, 교량, 도로, 수송관 등의 기반시설의 안전성에 영향을 주는 요소이다. 본 연구에서는 검사 비용과 시간을 줄일 수 있는 자동화된 균열 탐지 시스템을 다룬다. 환경과 표면에 강건한 시스템을 구성하기 위해서, 본 연구에서는 여러 사전 연구에서 사용된 다양한 표면의 균열 데이터 셋을 수집하여 통합 데이터 셋을 구축하였다. 이후, 컴퓨터 비전 분야에 높은 성능을 발휘하는 VGG, ResNet, WideResNet, ResNeXt, DenseNet, EfficientNet 딥러닝 모델을 적용하였다. 통합 데이터 셋은 훈련 집합(80%)과 테스트 집합(20%)으로 나누어 모델 성능을 검증하기 위해서 사용했다. 실험 결과, DenseNet121 모델이 높은 마라미터 효율성을 가지면서도 테스트 집합에 대해 96.20%의 정확도를 달성하여 가장 높은 성능을 보여주었다. 딥러닝 모델의 균열 검출 성능 증진을 통해, DenseNet121을 활용하여 컴퓨팅 자원이 적은 소형 디바이스에서도 높은 균열 검출 성능을 보이는 탐지 시스템을 구축이 가능함을 확인했다.

키워드 : 표면 검사, 균열 탐지, 컴퓨터 비전, 딥러닝

## 1. 서론

각종 건축물에서 발생하는 균열은 해당 건축물의 노후화를 촉진시킬 뿐만 아니라 장기적으로 안전성을 크게 저하시키는

위험 지표이다[1]. 따라서 균열을 조기에 발견하여 대처하는 것은 건축물의 안전성 유지에 있어 중요하다[2]. 기존의 건축물 안전점검은 경험과 기술을 갖춘 검사자가 육안으로 시설물의 손상이나 결함을 식별하는 것으로 수행되고 있다. 이와 같은 점검 방식은 검사자의 주관성과, 숙련도, 피로도 등에 따라 검사자의 신뢰도에 큰 영향을 끼치며 많은 비용과 시간이 소요된다.

이에 따라 균열 탐지와 관련하여 자동화된 검사 시스템의 필요성이 대두되고 있다[3]. 균열 탐지는 크게 표면 균열 탐지와 내부의 균열 탐지로 구분할 수 있다. 이 중 표면 균열을 탐지하기 위해 컴퓨터 비전 기술이 적용되고 있다. 건축물 균열 탐지 분야에서 영상에서 질감 등을 추출하는 Gabor 필터와 LBP(Local Binary Pattern), 직선의 요소인 엣지와 같은

\* 이 논문은 한국건설기술연구원 주요사업(지하매설 압력관의 실시간 건전성 진단 및 관리 기술 개발)의 지원을 받아 수행된 연구결과이며 이에 감사드립니다.

† 준회원: 성균관대학교 인공지능융합학과 석사과정

†† 비회원: 성균관대학교 인터랙션사이언스 석사

††† 비회원: 성균관대학교 인터랙션사이언스 석사과정

†††† 비회원: 성균관대학교 인공지능융합학과 석사과정

††††† 비회원: 성균관대학교 인터랙션사이언스학과, 인공지능융합학과 조교수

†††††† 비회원: 한국건설기술연구원 인프라안전연구본부 선임연구위원

††††††† 비회원: 한국건설기술연구원 차세대 인프라연구센터 선임연구위원

Manuscript Received : October 14, 2020

Accepted : November 14, 2020

\* Corresponding Author : Park Eunil(eunilpark@skku.edu)

특징을 추출하는 Canny 검출기를 통한 연구가 진행되었다 [3-5]. 또한 최근 급격히 발달하고 있는 딥러닝 기법을 표면 균열 검출에 적용하려는 연구도 수행되고 있다[9-11]. 이러한 연구들은 이미지 인식과 분류 문제에 합성곱 신경망을 적용하는 방법을 제시하였다.

다만 기존의 연구들에는 몇 가지 한계점이 있다. 먼저, 균열은 다양한 표면과 환경에서 발생할 수 있는데 비해 위 연구들은 하나의 데이터 셋에 대해서만 연구를 수행하였다. 또한, 상대적으로 적은 수의 딥러닝 모델을 기반으로 성능 비교를 수행했다. 본 연구에서는 다양한 표면과 환경에서 적용 가능한 일반적인 균열 탐지 모델을 구성하고자 한다. 이와 함께, 이미지 분류 문제에 있어 높은 성능을 발휘하는 여러 모델을 사용하여 균열 탐지에 있어 최적의 성능을 내는 모델을 탐구하고자 한다.

본 논문의 구성은 2장에서는 표면 균열 검출 관련 선행 연구를 설명하고 3장에서는 데이터 셋과 전처리, 딥러닝 모델을 포함한 연구방법을 제시하였으며, 4장에서는 모델 검증, 5장에서는 결론과 향후 연구 방향에 대하여 기술하였다.

## 2. 관련 연구

### 2.1 표면 균열 검출 방법

컴퓨터 비전 기술을 활용한 균열 검출 시스템을 제시한 연구가 진행되었다. 먼저, [4]는 이웃하는 픽셀의 크기 비교를 통해 영상에서 질감(texture)을 추출하는 LBP(Local Binary Pattern)를 사용하여 포장도로의 표면 균열을 탐지하였다. 또한, [3]은 코사인과 가우시안 함수를 통해 구성된 Gabor 필터로 포장도로의 표면 균열을 탐지하여 95%의 높은 정밀

도(Precision)를 달성하였다. [5]은 Canny 검출기와 전처리 기법을 통해 균열을 검출하였다.

그러나 위와 같은 접근법은 연구자가 적용 영역에 따라 특징 추출 방법을 설계해야 하는 번거로움과 해당 영역 이외의 영역에서는 성능차이가 크게 나타나는 문제점이 존재한다. 최근 딥러닝 기법을 활용한 컴퓨터 비전 연구가 활발하게 진행되고 있어, 데이터를 기반으로 의미 있는 특징을 학습함으로써 기존의 컴퓨터 비전 기법보다 더 높은 성능을 보인다 [6-8]. 딥러닝 모델인 합성곱 신경망은 회선(convolution) 연산을 통해 이미지에서 저수준과 고수준의 특징을 학습함으로써 높은 인식과 분류 성능을 발휘한다.

이에 딥러닝을 활용하여 표면 균열을 검출하는 연구들이 다수 수행되고 있다. [9]에서는 4개의 합성곱 층을 가진 신경망을 제안하여 가로 99, 세로 99 픽셀의 균열과 정상 이미지를 분류하도록 학습하였다. 해당 신경망은 정밀도(Precision) 86.96%와 재현율(Recall) 92.51% 그리고 F1-점수 89.65%를 기록하여 다른 SVM, Boosting 모델에 비해 높은 성능을 기록하였다. 또한, [10]에서는 8개의 층을 가진 합성곱 신경망 모델을 구성하여 콘크리트의 균열을 검출하였다. 모델은 약 4만 장의 가로 256, 세로 256 픽셀의 이미지를 입력받아 98%의 높은 정확도를 달성하였다. [11]에서는 유명 합성곱 신경망인 VGG16, InceptionV3, ResNet 모델을 사용하여 콘크리트의 균열을 검출하는데 적용하였다. 모델은 가로 227, 세로 227의 이미지를 입력받아 균열 여부를 탐지하였으며 VGG16 모델이 약 99.9%의 가장 높은 정확도를 보였다.

이에 본 연구에서 콘크리트, 포장도로, 육교 등의 다양한 건물물 표면의 균열 데이터를 수집하고 이를 통해 다양한 표면에서도 적용이 가능한 균열 검출 모델을 제시하고 검증하고자 한다.

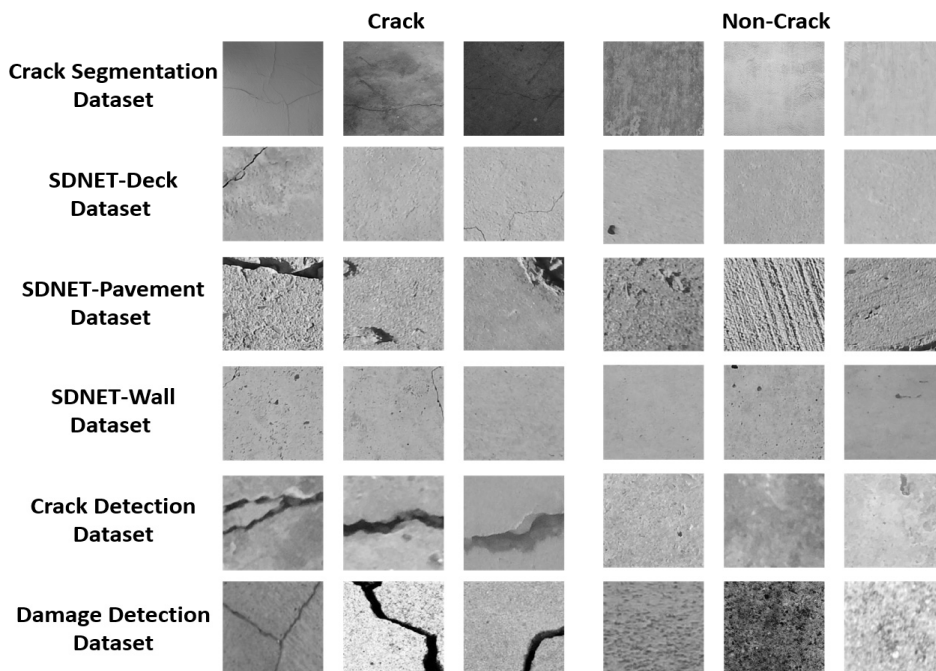


Fig. 1. Example of Employed Datasets

### 3. 실험 방법

#### 3.1 데이터 셋

본 연구에서는 다음과 같이 균열과 다양한 균열 탐지를 위한 접근을 수행하였다. Fig. 1은 사용된 데이터 셋의 일부분을 보여준다. 첫째로, 균열 탐지 및 분석을 위한 데이터 셋을 수집하였다[21]. 본 연구에서는 Crack Segmentation 데이터 셋으로 명명한다. 이 데이터 셋은 본래 균열의 위치를 판별하기 위한 영상 분할(semantic segmentation)을 위해 고안된 데이터 셋으로 포장도로와 콘크리트 구조물의 표면 균열 이미지와 정상 이미지, 그리고 균열의 위치를 표시하고 있는 마스크 이미지를 포함하고 있으며 각 이미지는 가로 448, 세로 448 픽셀로 구성되어있다. 본 연구에서는 해당 데이터 셋의 표면 균열 이미지와 정상 이미지 9,887장과 1,411장을 각각 사용하였다.

둘째로, 기존의 연구에서 활용한 균열 관련 데이터 셋을 활용하였다[22]. 이 데이터 셋은 54개의 교량 갑판, 104 개의 포장도로, 72개의 벽에서 추출한 균열 이미지와 정상 이미지를 포함하여 약 56,000장의 이미지를 포함하고 있다. 본 연구에서는 각 데이터 셋을 SDNET-Deck, SDNET-Pavement, SDNET-Wall 데이터 셋으로 명명한다. 각 이미지는 가로 256, 세로 256픽셀이며 추출된 장소별로 균열 이미지와 정상 이미지는 교량 갑판에서 2,025장, 11,595장, 포장도로에서 2,608장, 21,726장 그리고 벽에서 3,851장, 14,287장이다. 각 이미지는 그림자와 거칠기, 구멍 및 파편 등의 다양한 잡음(noise)을 포함하고 있다.

셋째로, 표면 균열 탐지에 관련한 데이터 셋이다.[23] 본 연구에서는 Crack Detection 데이터 셋으로 명명한다. 이 데이터 셋은 다양한 콘크리트 표면의 가로와 세로가 각각 227 픽셀인 균열 이미지 20,000장과 정상 이미지 20,000장을 포함하고 있다.

마지막으로, 피해손상 탐지 분야에서 기존에 활용되었던 데이터 셋이다[10]. 본 연구에서는 Damage Detection 데이터 셋으로 명명한다[24]. 해당 데이터 셋은 조명, 그림자 등이 포함된 건물의 가로와 세로가 각각 128 픽셀의 표면 균열 이미지와 정상 이미지를 각각 991장과 1,264장을 포함하고 있다.

본 연구에서는 위의 데이터 셋에서 각각 80%의 이미지를 추출하여 모델의 훈련 데이터로 사용하고 남은 20%의 데이터로 모델의 성능을 검증하는 테스트 데이터로 사용하였다.

각 이미지는 가로 224, 세로 224 픽셀로 조정되었고 0~255의 값을 가지던 각 픽셀을 0과 1 사이로 변환하였다.

#### 3.2 모델 구성

본 연구에서는 이미지 분류 문제에 높은 성능을 보이고 있는 다양한 딥러닝 모델을 사용하여 균열 분류 성능을 비교하였다. 본 연구에서는 VGG16, 19[12], Residual Network (Res) 34, 50, 101, 152[13], Dense Network (Dense) 121, 161, 169, 201[14], Wide Residual Network (WRes) 50,

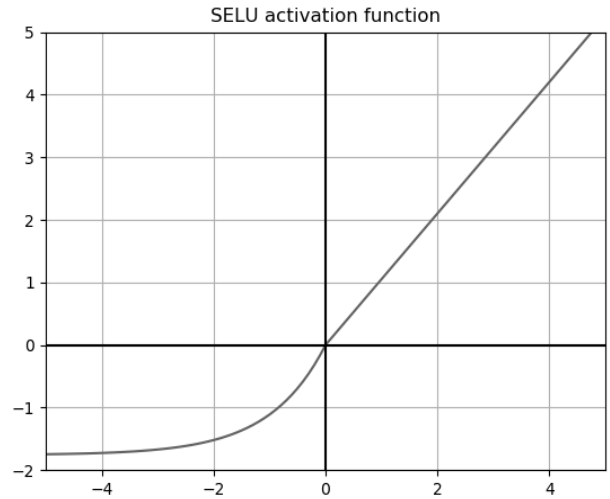


Fig. 2. SELU Activation Function

101[15], ResNeXt (ResNX) 50, 101[16], EfficientNet (Eff) B0, B2, B4[17]를 사용하였다.

높은 분류 성능을 달성할 수 있도록 각 모델마다 작은 변형을 주었다. 먼저, 각 모델에서 배치 정규화나 그룹 정규화 등의 정규화 계층과 함께 사용된 합성곱 계층은 Weight Standardization 이 적용된 합성곱 계층으로 교체되었다. 이에 따라 배치 정규화가 적용되지 않았던 VGG 모델에는 합성곱 계층 이후에 배치 정규화 계층을 추가하였다. 또한, EfficientNet 모델은 Swish와 Sigmoid 활성화 함수를 사용하여 구현하였으며 그 이외의 모델은 SELU(Scaled Exponential Linear Unit) 활성화 함수를 사용하여 모델을 구현하였다. 모델의 출력층은 균열이미지와 정상이미지 분류를 위해 softmax 함수와 2개의 뉴런을 가지는 층으로 교체되었으며 모델의 손실함수로 cross entropy 함수가 적용되었다.

#### 1) Weight Standardization

정규화 층은 모델이 그래디언트가 소실(vanishing)되거나 폭발(exploding)하는 것을 방지하여 모델의 최적화 과정이 안정적으로 진행되도록 한다. Weight Standardization(WS)은 합성곱 층의 가중치에 대해 정규화 과정을 수행함으로 배치 정규화가 가지는 배치 크기에 의존적인 특성을 개선하여 모델이 효율적으로 학습하도록 한다.[18] WS는 배치 정규화, 그룹 정규화 등의 정규화 층과 함께 사용된다. Equation (1)은 WS를 적용한 합성곱 층의 연산을 나타내며,  $y$ 는 층의 출력,  $\hat{w}$ 는 층의 가중치  $w$ 에 WS를 적용한 가중치, \*는 합성곱 연산,  $x$ 는 이전 층에서 전달된 입력이다.

$$y = \hat{w} * x \tag{1}$$

Equation (4)에서  $\hat{w}$ 는 [출력 차원, 1] 크기를 만족하는 행렬,  $I$ 는 층의 입력 채널  $\times$  커널의 크기,  $\hat{w}_{i,j}$ 는 행렬에서  $i, j$  번째

Table 1. Accuracy of Each Model for the Datasets

Model	Crack Segmentation Dataset (%)	SDNET-Deck Dataset (%)	SDNET-Pavement Dataset (%)	SDNET-Wall Dataset (%)	Crack Detection Dataset (%)	Damage Detection Dataset (%)	Combined Dataset (%)
VGG16	97.26	90.31	93.20	89.64	99.31	89.14	94.82
VGG19	97.17	90.53	93.38	89.94	99.29	90.02	94.93
ResNet34	98.50	90.64	93.73	90.93	99.74	94.90	95.59
ResNet50	98.23	89.06	93.59	89.42	99.63	93.35	95.01
ResNet101	98.50	91.59	94.06	91.21	99.74	97.18	95.87
ResNet152	<b>98.76</b>	89.68	94.04	90.99	99.61	95.79	95.55
ResNeXt50	97.92	90.68	93.77	90.55	99.75	94.90	95.49
ResNeXt101	<b>98.76</b>	91.45	94.82	91.79	99.66	93.13	96.04
WideResNet50	98.50	91.74	94.68	91.32	99.65	95.79	95.99
WideResNet101	98.32	91.30	94.53	90.55	99.58	92.68	95.66
DenseNet121	98.41	91.85	94.84	91.92	99.78	96.90	<b>96.20</b>
DenseNet161	98.45	91.63	94.68	<b>92.39</b>	<b>99.83</b>	94.01	96.17
DenseNet169	98.50	91.56	<b>95.07</b>	91.70	99.69	<b>97.34</b>	96.16
DenseNet201	98.50	<b>91.92</b>	94.41	90.93	99.80	94.90	95.92
EfficientNetB0	98.27	91.15	93.88	89.50	99.45	93.57	95.29
EfficientNetB2	98.05	90.27	93.86	89.44	99.53	96.01	95.26
EfficientNetB4	98.05	87.41	93.08	88.12	99.40	96.23	94.44

요소를 나타낸다. WS를 적용하는 은닉층에서는 Equation (2)와 Equation (3)을 통해서 커널 별로 평균  $\mu_{w_{i,j}}$  과 표준편차  $\sigma_{w_{i,j}}$ 를 구한 후에 Equation (4)를 통해 정규화를 수행한다. 모델은 WS를 적용하기 이전의 파라미터  $\mathbf{w}$ 에 대해 손실 함수의 그래디언트를 계산해 최적화 된다.

$$\mu_{w_{i,j}} = \frac{1}{I} \sum_{j=1}^I w_{i,j} \quad (2)$$

$$\sigma_{w_{i,j}} = \sqrt{\frac{1}{I} \sum_{j=1}^I w_{i,j}^2 - \mu_{w_{i,j}}^2 + \epsilon} \quad (3)$$

$$\hat{\mathbf{w}} = [\hat{w}_{i,j} \mid \hat{w}_{i,j} = \frac{w_{i,j} - \mu_{w_{i,j}}}{\sigma_{w_{i,j}}}] \quad (4)$$

## 2) Scaled Exponential Linear Unit

SELU 함수는 ELU 함수에 scale 파라미터가 추가된 활성화 함수이다.[19] 딥러닝 모델에서 널리 사용되는 ReLU 함수는 최적화시에 그래디언트가 소실되는 문제를 해결하여 효과적으로 손실함수의 최적점을 찾아가도록 한다. 그러나 ReLU

함수는 음의 입력 값에 대한 미분 값이 0이므로 음수가 되어 버린 뉴런이 다시 활성화 되지 않는 죽은 ReLU (dead-ReLU) 문제를 가지고 있다. SELU 함수는 ReLU 함수처럼 양의 입력 값에 대해 그대로 값을 출력함으로써 모델이 최적점을 효과적으로 찾아가도록 도울 뿐만 아니라 음수에 대해서도 미분 값이 남아 있어 모델이 효과적으로 학습하도록 한다. SELU 수식은 Equation (5)와 같으며  $\lambda$ 는 1.0507,  $\alpha$ 는 1.6732 로 설정하였다.

$$f(\alpha, x) = \lambda \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases} \quad (5)$$

## 3) 모델 최적화

모델을 효과적으로 학습하기 위해서 Adam 옵티마이저를 적용하였다.[20] 초기 학습률  $\eta$ 은 0.0005로 설정하였으며  $\beta_1$ 과  $\beta_2$ 는 각각 0.9와 0.999로 설정하였다. 또한 모델 가치가 손실함수의 최적점을 지나쳐 학습이 제대로 진행되지 못할 것을 방지하기 위해 학습률 감소 스케줄링을 적용하였다. 본 연구에서는 한 에폭 마다 이전 학습률  $\eta$ 에 scale fac

Table 2. Number of Parameters and Ratio of Each Model

Model	Number of Parameters	Parameters Ratio (%)
VGG16	138,365,992	96.30
VGG19	143,678,248	100.00
ResNet34	21,797,672	15.17
ResNet50	25,557,032	17.79
ResNet101	44,549,160	31.01
ResNet152	60,192,808	41.89
ResNeXt50	25,028,904	17.42
ResNeXt101	88,791,336	61.80
WideResNet50	68,883,240	47.94
WideResNet101	126,886,696	88.31
DenseNet121	7,978,856	5.55
DenseNet161	28,681,000	19.96
DenseNet169	14,149,480	9.85
DenseNet201	20,013,928	13.93
EfficientNetB0	5,288,548	3.68
EfficientNetB2	9,109,994	6.34
EfficientNetB4	19,341,616	13.46

tor  $\gamma$ 로 0.96을 곱해 지수 감소 스케줄링을 적용하였다. 모델은 훈련 데이터에 대해 20 에폭 학습하였으며 배치 사이즈는 32로 통일하여 학습하였다.

$$\eta_{m+1} := \gamma \times \eta_m \tag{6}$$

### 3.3 성능 검증

다양한 표면에 대한 모델의 균열 탐지 성능을 확인하기 위해서 훈련 데이터에 대해서 학습된 모델을 테스트 데이터에 대해서 적용하였다. 이때, 다양한 표면과 환경에서의 모델의 균열 탐지 성능을 검증하기 위해 각 데이터 셋에 대한 성능

검증과 전체 데이터에 대한 성능 검증을 모두 수행하였다. 즉, Crack Segmentation 데이터 셋, SDNET-Deck 데이터 셋 등 각 데이터 셋에서 테스트 데이터 각각에 대한 성능 검증과 이들 테스트 데이터 모두를 활용한 통합 데이터 셋 (Combined Dataset)에 대한 성능 검증을 모두 수행하였다.

### 4. 실험 결과

모델의 각 데이터 셋에 대한 분류 정확도는 Table 1에 나타나 있다. 각 모델의 정확도를 소수점 3자리에서 반올림하여 표시하였다. 전반적으로 DenseNet 모델이 높은 성능을 보여주고 있다. 통합 데이터 셋에서 DenseNet121이 96.20%로 가장 높은 정확도를 보여주고 있다. Crack Segmentation 데이터 셋에 대해서는 ResNet152와 ResNeXt101이 각각 98.76%로 높은 정확도를 기록하였다. SDNET-Deck 데이터 셋에 대해서는 DenseNet201이 91.92%, SDNET-Pavement 데이터 셋에 대해서는 DenseNet169이 95.07%, SDNET-Wall 데이터 셋에 대해서는 DenseNet161이 92.39%로 각각 가장 높은 정확도를 달성하였다. 한편, Crack Detection 데이터 셋에 대해서는 DenseNet161이 99.83%, Damage Detection 데이터 셋에 대해서는 DenseNet169가 97.34%로 가장 높은 정확도를 달성하였다. 이를 통해 각 데이터 셋에 대해서는 여러 모델들이 각기 높은 성능을 발휘 하였으나 DenseNet121 모델이 전체 데이터에 대한 일반화 성능이 가장 높은 것을 확인할 수 있다.

Table 2는 각 모델의 파라미터 숫자와 가장 많은 수의 파라미터를 가진 모델인 VGG19 대비 모델의 파라미터 수의 비율을 보여주며 Fig. 3은 파라미터 수와 정확도 간의 비교를 위해 시각화한 그래프이다. 일반적으로 파라미터가 많을수록 모델의 용량(capacity)이 늘어나 학습할 수 있는 능력이 높아지나 모델이 과적합 되어 훈련이 어려워지거나 소형 장비 등에 배포하기 어려워지는 등의 문제점이 존재한다. 본 연구에서 사용된 모델 중 가장 많은 파라미터를 가지고 있는 모델은 VGG19로 143,678,248개의 파라미터를 가지고 있다. 그러나 통합 데이터 셋에서 가장 높은 성능을 보여주는 DenseNet121은

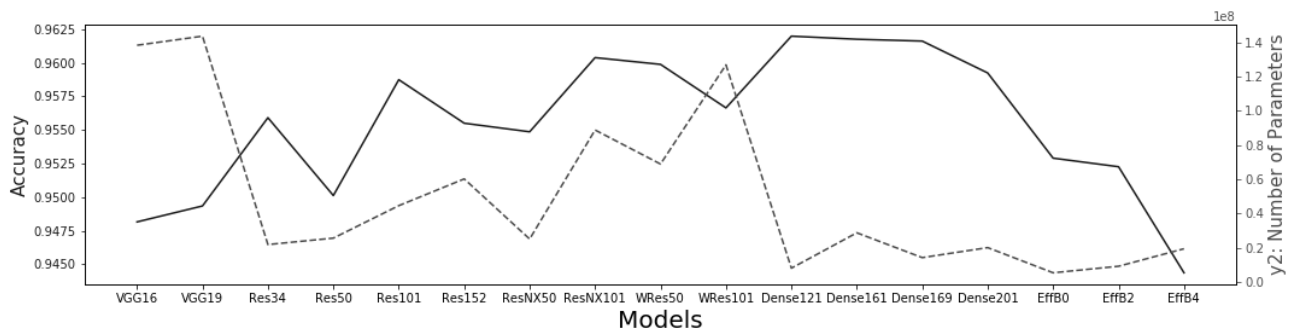


Fig. 3. Comparison between Accuracy and Number of Parameters of Each Model

VGG19의 약 5.55%인 7,978,856개의 모델 파라미터를 가지고도 가장 높은 성능을 보인다. 뿐만 아니라 각 데이터 셋에 대해 높은 성능을 보이는 모델들 중에서도 DenseNet 모델의 파라미터 비율은 상당히 적다. ResNet152와 ResNeXt101의 모델 파라미터가 각각 60,192,808, 88,791,336개로 VGG19 대비 파라미터 비율이 41.89%, 88.31% 인데 반하여 DenseNet161, DenseNet169, DenseNet201의 파라미터 개수는 28,681,000, 14,149,480, 20,013,928 개로 각 비율이 19.96%, 9.85%, 13.93% 밖에 되지 않는다. 다른 모델에 비해서 DenseNet 모델들은 상대적으로 적은 파라미터 만을 가지고 효율적으로 균열을 탐지한 것을 확인할 수 있다.

## 5. 결 론

균열은 건축물의 안전성과 성능을 위협하는 위험 요소이므로 이를 조기에 탐지하여 대처하는 것은 안전성 유지에 있어 중요하다. 본 연구에서는 다양한 표면과 환경에 모두 적용 가능한 균열 탐지 모델을 구성하기 위해서 여러 표면과 환경에서 추출한 균열 데이터 셋을 통합하여 사용하였다. 또한, 효과적인 모델을 구성하기 위해서 이미지 분류 문제에 뛰어난 성능을 보이는 17개의 최신 딥러닝 모델에 Weight Standardization을 적용한 합성곱과 SELU 활성화 함수를 적용하여 구현하였다. 이후, 구성된 모델을 통합 데이터 셋에 학습한 후에 그 성능을 비교하고 검증하였다. 그 결과 DenseNet 기반의 모델이 각 데이터 셋에 대해 탁월한 성능을 발휘하였다. 특히 DenseNet121 모델이 통합 데이터 셋에 대해 VGG19 모델의 오직 5.55%의 파라미터만을 사용하여 가장 높은 정확도인 96.20%를 달성하였다. 이를 통하여 DenseNet 모델을 활용하여 일반화된 균열 탐지 시스템을 구성할 수 있을 뿐만 아니라 높은 파라미터 효율성으로 컴퓨팅 자원이 부족한 환경에서도 효율적으로 운용이 가능할 것으로 확인된다.

특히, 본 연구에서 적용한 Weight Standardization과 SELU 함수의 경우, 향후 균열 탐지와 이상상태 판별을 위한 최적화 기법으로 활용할 수 있는 이론적 근거를 마련하였다. 다만, 이를 일반화하여 활용하기 위해서는 각 변수와 함수를 자동으로 설정하기 위한 추가 방법론이 필요할 것으로 보인다.

본 연구를 통해 균열과 관련하여 높은 정확도를 추출하는 모델을 개발하였으나, 보다 다양한 균열의 종류를 다루지 못했다는 한계점이 존재한다. 뿐만 아니라, DenseNet 기반의 모델이 각 데이터 셋에 대해 높은 성능을 보였으나, 이는 수집된 균열의 특성과 종류에 따라 그 편차가 심한 것으로 나타나 이에 대한 충분한 고찰을 수행할 필요가 있다. 마지막으로 같은 모델에 따른 연구 결과를 도출했음에도 Damage Detection Dataset과 같이 특수한 경우에서 그 성능 지표의 차이가 많이 난 것을 볼 때, 추후 연구에서는 주변 환경과 기타 변수 조절을 통한 최적화에 보다 집중할 필요가 있을 것으로 보인다.

## References

- [1] B. Kim, G. Kim, S. Jin, and S. Cho, "A comparative study on performance of deep learning models for vision-based concrete crack detection according to model types," *Journal of the Korean Society of Safety*, Vol.34, No.6, pp.50-57, 2019.
- [2] J. Zhang, C. Lu, J. Wang, L. Wang, and X.-G. Yue, "Concrete cracks detection based on fcn with dilated convolution," *Applied Sciences*, Vol.9, No.13, pp.2686, 2019.
- [3] M. Salman, S. Mathavan, K. Kamal, and M. Rahman, "Pavement crack detection using the gabor filter," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, pp.2039-2044, 2013.
- [4] Y. Hu and C.-X. Zhao, "A novel lbp based methods for pavement crack detection," *Journal of pattern Recognition Research*, Vol.5, No.1, pp.140-147, 2010.
- [5] B. Santhi, G. Krishnamurthy, S. Siddharth, and P. Ramakrishnan, "Automatic detection of cracks in pavements using edge detection operator," *Journal of Theoretical and Applied Information Technology*, Vol.36, No.2, pp.199-205, 2012.
- [6] Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.580-587, 2014.
- [7] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, pp.1097-1105, 2012.
- [8] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*, Springer, pp.234-241, 2015.
- [9] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp.3708-3712, 2016.
- [10] Y.-J. Cha, W. Choi, and O. Büyüköztürk, "Deep learning-based crack damage detection using convolutional neural networks," *Computer-Aided Civil and Infrastructure Engineering*, Vol.32, No.5, pp.361-378, 2017.
- [11] C. V. Dung et al., "Autonomous concrete crack detection using deep fully convolutional neural network," *Automation in Construction*, Vol.99, pp.52-58, 2019.
- [12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.770-778, 2016.

[14] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.4700-4708, 2017.

[15] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *British Machine Vision Conference*, 2016.

[16] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1492-1500, 2017.

[17] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019.

[18] S. Qiao, H. Wang, C. Liu, W. Shen, and A. Yuille, "Weight standardization," *arXiv preprint arXiv:1903.10520*, 2019.

[19] G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter, "Self-normalizing neural networks," in *Advances in Neural Information Processing Systems*, pp.971-980, 2017.

[20] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, 2015.

[21] kxhaha, Crack Segmentation [Internet], [https://github.com/kxhaha/crack\\_segmentation#Dataset](https://github.com/kxhaha/crack_segmentation#Dataset)

[22] M. Maguire, S. Dorafshan and R. J. Thomas, SDNET2018: A concrete crack image dataset for machine learning applications [Internet], [https://digitalcommons.usu.edu/all\\_datasets/48/](https://digitalcommons.usu.edu/all_datasets/48/)

[23] Arun Pandian R, Surface Crack Detection: Concrete surface sample images for Surface Crack Detection [Internet], <https://www.kaggle.com/arunrk7/surface-crack-detection>

[24] satyenrajpal, Concrete-Crack-Detection [Internet], <https://github.com/satyenrajpal/Concrete-Crack-Detection>



**지 홍 군**

<https://orcid.org/0000-0002-5245-5686>  
 e-mail : jeehg1@skku.edu  
 2020년 한국산업기술대학교 컴퓨터공학과 (학사)  
 2020년~현재 성균관대학교 인공지능융합학과 석사과정

관심분야: Deep Learning for Multimedia



**김 지 나**

<https://orcid.org/0000-0003-1140-6113>  
 e-mail : jinakim@g.skku.edu  
 2019년 한성대학교 컴퓨터공학과(학사)  
 2021년 성균관대학교 인터랙션사이언스 (석사)  
 관심분야: Large Scale Machine Learning/Deep Learning



**황 시 정**

<https://orcid.org/0000-0002-9957-0796>  
 e-mail : gigr4132@naver.com  
 2020년 성균관대학교 경제학과(학사)  
 2020년~현재 성균관대학교 인터랙션사이언스 석사과정  
 관심분야: Data science & Movie Analysis



**김 도 건**

<https://orcid.org/0000-0002-0859-7492>  
 e-mail : dgkimk93@gmail.com  
 2018년 인천대학교 임베디드시스템공학과 (학사)  
 2020년~현재 성균관대학교 인공지능융합학과 석사과정

2018년~현재 한화시스템 사원  
 관심분야: Computer Vision, Deep Learning



**박 은 일**

<https://orcid.org/0000-0002-3177-3538>  
 e-mail : eunilpark@skku.edu  
 2010년 성균관대학교 전자컴퓨터공학과 (학사)  
 2012년 성균관대학교 인터랙션사이언스 (석사)

2016년 한국과학기술원(KAIST) 기술혁신(박사)  
 2015년~2017년 한국건설기술연구원(KICT) 전임연구원  
 2017년~2018년 한양대학교 ERICA 조교수  
 2018년~현재 성균관대학교 인터랙션사이언스학과, 인공지능융합학과 조교수  
 관심분야: 사용자경험, 데이터과학, 인공지능활용



**김 영 석**

<https://orcid.org/0000-0002-6398-3126>  
e-mail : kimys@kict.re.kr  
1999년 영남대학교 토목공학(학사)  
2002년 Osaka University 토목공학(지반)  
(석사)  
2005년 Kyoto University 토목공학(지반)  
(박사)

2005년 ~ 현 재 한국건설기술연구원 인프라안전연구본부  
선임연구위원  
관심분야: 극한지 건설기술



**류 승 기**

<https://orcid.org/0000-0002-2760-861X>  
e-mail : skryu@kict.re.kr  
1990년 충북대학교 전기제어계측(학사)  
1992년 충북대학교 전기제어계측(석사)  
1999년 충북대학교 전기공학(박사)

1994년 ~ 현 재 한국건설기술연구원 차세대 인프라연구센터  
선임연구위원  
관심분야: 지능형 교통기술