

<https://doi.org/10.7236/JIIBC.2020.20.1.163>

JIIBC 2020-1-23

## 딥 러닝 기반 실시간 센서 고장 검출 기법

# Timely Sensor Fault Detection Scheme based on Deep Learning

양재완\*, 이영두\*\*, 구인수\*\*\*

Jae-Wan Yang\*, Young-Doo Lee\*\*, In-Soo Koo\*\*\*

**요약** 최근 4차 산업혁명의 핵심기술인 인공지능, 빅데이터, 사물인터넷의 발전으로 산업 현장에서 가동되는 기계의 자동화 및 무인화에 대한 연구가 활발히 진행되고 있다. 이러한 공정 기계들은 부착된 다양한 센서들로부터 수집된 데이터를 기반으로 제어되고 이를 통해 공정이 관리된다. 만약 센서에 고장이 발생한다면 센서 데이터 이상으로 인해 자동화 기계들이 오작동함으로써 공정 손실 발생뿐만 아니라 인명피해로도 이어질 수 있다. 전문가가 센서의 이상 여부를 주기적으로 확인하여 관리하고 있으나 산업 현장의 여러 가지 환경요인 및 상황으로 인하여 고장점검 시기를 놓치거나 고장을 발견하지 못하여 센서 고장으로 인한 피해를 막지 못하는 경우가 발생하고 있다. 또한 고장이 발생하여도 즉각 감지하지 못함으로써 공정 손실을 더욱 악화시키고 있는 실정이다. 따라서 이러한 돌발적인 센서 고장으로 인한 피해를 막기 위해 자체적으로 임베디드 시스템에서 센서의 고장 유무를 실시간으로 파악하고 빠른 대응을 위해 고장 진단 및 유형을 판별하는 것이 필요하다. 본 논문에서는 대표적인 센서 고장 유형인 erratic fault, hard-over fault, spike fault, stuck fault를 분류하기 위해 딥 뉴럴 네트워크 기반의 고장 진단 시스템을 설계하고 라즈베리 파이를 활용하여 구현하였다. 센서 고장 진단을 위해 구글이 제안한 MobilieNetV2의 Inverted residual block 구조를 사용하여 네트워크를 구성하였다. 본 논문에서 제안하는 방식은 기존 CNN 기법을 사용한 경우보다 메모리 사용량이 줄고 성능이 향상되며, 입력 신호에 대해 구간별로 센서 고장을 분류하여 산업 현장에서 효과적으로 사용될 것으로 기대된다.

**Abstract** Recently, research on automation and unmanned operation of machines in the industrial field has been conducted with the advent of AI, Big data, and the IoT, which are the core technologies of the Fourth Industrial Revolution. The machines for these automation processes are controlled based on the data collected from the sensors attached to them, and further, the processes are managed. Conventionally, the abnormalities of sensors are periodically checked and managed. However, due to various environmental factors and situations in the industrial field, there are cases where the inspection due to the failure is not missed or failures are not detected to prevent damage due to sensor failure. In addition, even if a failure occurs, it is not immediately detected, which worsens the process loss. Therefore, in order to prevent damage caused by such a sudden sensor failure, it is necessary to identify the failure of the sensor in an embedded system in real-time and to diagnose the failure and determine the type for a quick response. In this paper, a deep neural network-based fault diagnosis system is designed and implemented using Raspberry Pi to classify typical sensor fault types such as erratic fault, hard-over fault, spike fault, and stuck fault. In order to diagnose sensor failure, the network is constructed using Google's proposed Inverted residual block structure of MobilieNetV2. The proposed scheme reduces memory usage and improves the performance of the conventional CNN technique to classify sensor faults.

**Key Words** : Sensor fault diagnosis, Convolution neural network, Inverted residual block, Raspberry pi

\*준회원, 울산대학교 전기공학부

\*\*정회원, 울산대학교 전기공학부

\*\*\*정회원, 울산대학교 전기공학부(교신저자)

접수일자: 2019년 11월 22일, 수정완료: 2020년 1월 2일

게재확정일자: 2020년 2월 7일

Received: 22 November, 2019 / Revised: 2 January, 2020 /

Accepted: 7 February, 2020

\*\*\*Corresponding Author: iskoo@ulsan.ac.kr

Dept. of Electrical Electronic Engineering, University of Ulsan, Korea

## I. 서 론

최근 4차 산업혁명의 핵심기술인 인공지능, 빅데이터, 사물인터넷과 같은 기술들이 제조 산업 전반으로 적용되어 기계들의 자동화 및 무인화시키는 것과 동시에 생산 효율을 최대화하는 스마트 팩토리 도입이 진행되고 있다. 스마트 팩토리에서는 생산성, 효율성, 에너지 절감 등을 고려하여 자동화된 생산 공정라인에서 신속정확하게 제품을 생산하는 다품종 고객맞춤형 생산체제를 갖추는 것이 핵심이다<sup>[1-2]</sup>. 생산체계가 대량생산에서 다품종 고객맞춤형 생산으로 변화하면서 단순한 공정을 반복하는 작업에서 사물인터넷과 인공지능 기술을 기반으로 하는 고도의 기술이 필요한 작업으로 발전하고 있다. 공장 내 기계와 설비에 부착된 다양한 센서들은 실시간으로 데이터를 수집하고 분석하여 이를 기반으로 기계의 구동상태 및 제어량을 파악하고 전체 공정라인의 작업 속도와 양을 조정하여 생산을 관리한다. 따라서 센서에 결함이 생길 경우, 데이터 오류로 인한 기계들의 오작동으로 공정 운영에 경제적 손실이 발생하는 것은 물론이고 인명피해로도 이어질 수 있다. 전문가가 일정 주기를 가지고 센서의 이상 여부를 확인하여 관리하지만 고장 점검 시기를 놓치거나 발견하지 못하는 경우도 있으며, 스마트 공장의 경우 방대한 양의 데이터를 수집하기 위해 센서의 양이 기하급수 적으로 증가할 것이므로 이를 모두 확인하는 것은 불가능하다. 따라서 자체적으로 임베디드 시스템에서 센서의 고장 유무를 실시간으로 파악하고 빠른 대응을 위해 고장 진단 및 유형을 판별하는 것이 필요하다.

스마트 팩토리에서는 고장 진단을 위해 다양한 센서를 사용하여 데이터를 수집, 분석한 뒤 고장 진단 알고리즘을 통해 고장 여부를 판별하는 시스템이 구축된다. 취득한 신호 데이터는 일반적으로 복잡한 작업환경으로 인해 잡음이 포함된다. 또한 고차원이므로 원신호를 사용하는 것보다 다양한 신호 처리 기술을 거쳐 고장의 검출 및 분류에 적용된다. 측정 신호로부터 평균, 표준편차, 첨도, 왜도와 같은 통계적 특징들을 추출하여 사용하거나 주파수 영역에서 푸리에 변환을 사용한 주파수 특징 추출, 웨이블릿 변환과 같은 방법들이 이상 감지 및 진단을 위한 특징으로 제안되었다<sup>[4-6]</sup>.

고장 신호를 검출 및 분류하기 위해 주어진 모델을 기반으로 신호 처리 및 분석, 규칙기반 전문가시스템, 퍼지 이론들을 적용한 연구가 활발히 이루어져 왔다. 하지만 이러한 방법들은 새로운 고장 유형이 발생할 경우 판별을 위한 임계값 설정과 같은 대처 알고리즘에 대한 추가

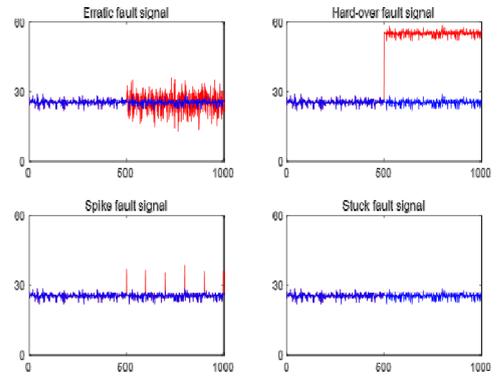


그림 1. 정상 신호(파랑)와 센서 고장 신호(빨강)  
Fig. 1. Normal signal(blue) and sensor fault signal(red)

및 수정이 어렵고 작업 환경의 복잡성으로 인해 일부의 방법들만 적용이 가능하다. 최근에는 이러한 단점을 보완하기 위해 Support Vector Machine(SVM)과 같은 기계학습이나 Convolutional Neural Network(CNN)와 같은 딥 러닝 기법이 고장 진단시스템에 적용되고 있으며, 성능 또한 우수하여 여러 분야에서 응용되고 있다<sup>[4-9]</sup>.

본 논문에서는 센서를 통해 데이터를 수집하고 라즈베리 파이 자체 내에서 센서 고장을 실시간으로 판별하고 그 결과를 서버로 보내어 사용자에게 알려주는 시스템을 제안한다. 구글에서 제안한 MobileNetV2의 Inverted residual block 구조를 사용하여 네트워크를 구성하고 이를 사용하여 대표적인 센서 고장 유형 4가지인 erratic, hard-over, spike, stuck에 대한 분류한 진단 결과를 분석 및 기술하였다. 제안된 네트워크와 기존 CNN모델 방식의 실험결과를 통해 성능을 비교·분석하였으며, 입력 신호에 대한 구간별 고장 진단의 성능 실험 결과도 제시하였다.

## II. 센서 고장 진단 시스템과 고장 신호

### 1. 센서 고장 진단 시스템 구성

본 논문에서 제안하는 센서 고장 진단 시스템은 그림 2와 같이 센서, 라즈베리 파이, 서버용 PC, DB로 구성된다. 센서를 통해 데이터를 측정하고 라즈베리 파이에서 수집된 데이터와 동시에 라즈베리 파이 자체에 Keras를 설치하여 딥 뉴럴 네트워크로 고장 판별 결과를 서버용 PC로 송신한다. 서버용 PC에서는 센서 데이터와 분류 결과를 DB에 저장하고 관리할 수 있다.

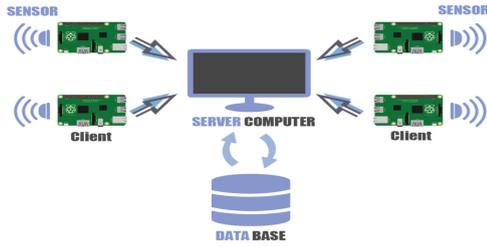


그림 2. 센서 고장 진단 시스템 구성도  
 Fig. 2. Structure of sensor fault diagnosis system

## 2. 센서 고장 데이터

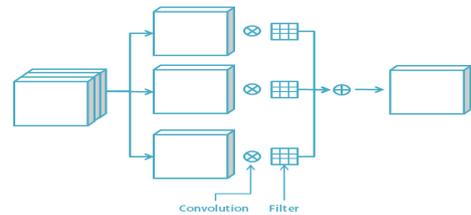
본 논문에서는 대표적인 센서 고장 유형 4가지인 erratic fault, hard-over fault, spike fault, stuck fault 데이터를 사용하였다<sup>[5]</sup>. 센서의 고장 신호와 정상 신호를 동시에 출력한 상태를 그림 1과 같이 나타내었다. Erratic fault는 센서 출력이 정상 신호에 대해 급격하게 노이즈가 증가된 형태를 보이며, 센서 출력의 분산이 크게 증가한다. Hard-over fault의 경우 센서 출력이 한계치를 넘어 biased되는 현상을 의미하며, 그림 1에서 보듯이 정상 신호의 값이 일정 기간 증가한 것을 볼 수 있다. Spike fault는 일정 간격으로 센서 출력 값이 튀는 (Spike) 현상이다. 마지막으로 stuck fault는 고정된 출력이 지속되는 것을 말한다. 센서 고장 신호의 특징들은 표 1과 같이 수식으로 표현할 수 있다. 수식에서 공통적으로  $x_n^c$ 는 n번 째 c유형의 신호를 의미한다. Erratic에서  $\dot{s}_n$ 는  $N(0, \delta^2)$ 의 정규분포를 따르는 노이즈를 의미한다. Hard-over와 spike에서 b 만큼 출력값이 증가하며 bias를 의미하는데, spike에서는 일정구간 n마다 발생한다. Stuck에서 a는 n지점에서 이후 고정된 출력 값을 의미한다.

본 논문에서 센서 고장 검출 및 분류를 위해 활용된 데이터는 TC1047/TC1047A Precision Temperature-to-Voltage Converter로부터 측정된 데이터와 표 1에서 정의된 수식을 기반으로 한다<sup>[5]</sup>. Normal, erratic fault, hard-over fault, spike fault, stuck fault는 1,000개의 데이터 샘플들로 구성되며, 총 5,000개의 데이터 샘플을 사용하였다. 데이터 샘플 하나는 1,000개의 데이터 포인트를 가지며 sampling rate는 25Hz이다.

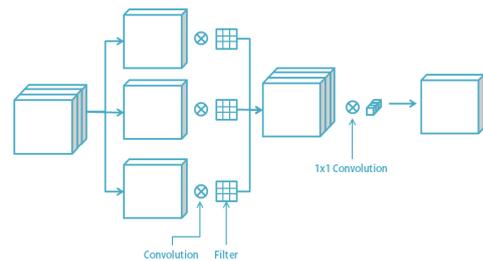
표 1. 센서 데이터 고장 유형 및 수학적 표현  
 Table 1. Sensor data fault type and mathematical expression

Fault type	Mathematical Expression of Fault
Erratic	$x_n^{erratic} = x_n^{normal} + \dot{s}_n, \dot{s}_n \sim N(0, \delta^2)$
Hard-over	$x_n^{hardover} = x_n^{normal} + b, b = constant$
Spike	$x_n^{spike} = x_n^{normal} + b_n, b_n = constant$
Stuck	$x_n^{stuck} = a, a = constant$

## III. 센서 고장 진단을 위한 딥 뉴럴 네트워크



(a) Regular convolution 과정



(b) Depthwise separable convolution 과정

그림 3. Convolution 연산 과정 비교  
 Fig. 3. Comparison of convolution operation

최근 CNN이 ILSVRC(ImageNet Large Scale Visual Recognition Challenge)나 Microsoft COCO(Common Object in Context)와 같은 국제 대회에서 우수한 성능을 보여주었다<sup>[10-13]</sup>. 하지만 VGG, ResNet과 같은 모델들은 깊은 네트워크 구조를 가지며, 이에 따라 연산량과 파라미터 수가 많아 모바일과 임베디드 환경에서 사용되기에는 적합하지 못하다. 이를 개선하기 위해 구조를 간소화시켜 연산 효율을 높이고 성능은 떨어지지 않는 MobileNet, Xception과 같은 모델들이 제안되었다<sup>[14-16]</sup>.

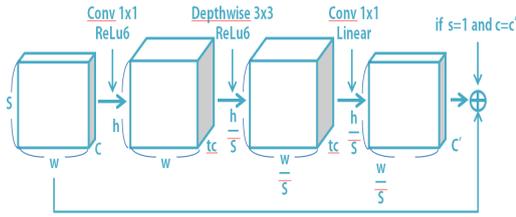


그림 4. Inverted residual 구조  
Fig. 4. Inverted residual structure

### 1. Depthwise separable convolution

CNN에서 일반적으로 사용되는 convolution 연산과 MobileNet에서 사용된 depthwise separable convolution 연산 과정을 그림 3에 각각 나타내었다. 일반적인 convolution 과정에서는 이전 input 채널과 동일한 개수의 필터 채널을 사용하여 공간 방향과 채널 방향으로 한 번에 더하여 연산을 진행한다. 반면, depthwise separable convolution의 경우, 공간 방향에 대한 depthwise convolution과 채널 방향에 대한 pointwise convolution으로 수행된다. 우선 그림 3에 (b)처럼 depthwise convolution을 사용하여 특징맵의 각 채널에 대해 공간 방향의 convolution 연산을 수행한 뒤, pointwise convolution으로 채널 방향의 1x1 convolution을 진행한다. 이와 같이 공간과 채널 방향의 convolution을 분리하여 계산하는 방식이 기존의 convolution보다 적은 parameter를 사용하고 연산량을 줄이는 효과를 얻게 한다. MobileNet<sup>[14]</sup>에서 N을 출력 채널, k를 필터 사이즈라고 할 때  $(\frac{1}{N} + \frac{1}{k^2})$ 배 정도 연산량을 줄일 수 있으며 기존의 네트워크 모델에 비해 약 8~9배정도 속도가 개선되었다.

### 2. Inverted residual

본 논문에서 네트워크를 구성하기 위해 사용된 inverted residual 구조는 그림 4와 같다<sup>[16]</sup>. 입력이 주어지면 1x

1 convolution과 depthwise convolution을 통과하면서 내부 채널을 확장시킨 후 다시 원상태로 줄여 입력과 더하는 방식이다. 그림 5에서 32의 feature map의 경우, 6배(t=6)인 192로 확장 뒤 입력과 더하고 3번(n=3) 반복해서 이러한 구조를 사용한다. 이와 같은 방식은 기존에 ResNet에서 사용된 residual block 방식에 비해 더 적은 입력과 출력 파라미터가 필요하므로 메모리 효율성이 우수하다<sup>[16]</sup>. 또한 linear층을 사용함으로써 입력에 대한 정보 보존이 가능하다.

### 3. 제안한 딥 뉴럴 네트워크 구조

본 논문에서 센서 고장 진단을 위해 제안한 딥 뉴럴 네트워크 모델의 구조는 그림 5에 나타난 것과 같다. 센서 데이터의 입력은 총 1,000개의 데이터 포인트를 사용하였으며 convolution 연산을 위해 40x25 행렬 형태로 사용되었다. 입력에서 한 행은 초당 수집한 데이터를 의미한다. 이를 3x3 convolution 필터, stride를 (1, 4)로 연산하여 32개 feature map을 얻는다. 이후 mobilenet V2의 핵심인 inverted residual bottleneck을 3번 쌓아 사용하였으며, 3x3 convolution 필터를 사용하여 16, 24, 32로 feature map을 추출하였다. 확장 계수(t)는 각각 1, 6, 6이고 inverted residual 구조 반복 계수(n)는 1, 2, 3이다. 이후 첫 convolution 층과 같은 방식으로 128 feature map을 추출하고 max pooling과 1x1 convolution을 통해 하나의 데이터 샘플 입력에 대해 40x1의 출력을 가진다. 최종적으로 softmax를 거쳐 센서 고장을 판별한다.

## IV. 실험 및 결과

본 논문에서 딥러닝 학습 및 테스트를 위해 사용된 센서 데이터는 클래스별 1,000개씩 총 5,000 샘플들이 사용되었다. 이 중에서 학습을 위해 3,000개의 샘플들이

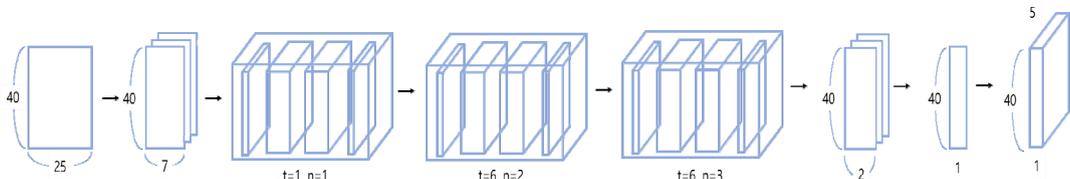


그림 5. 센서 고장 진단을 위한 딥 뉴럴 네트워크 모델 구조  
Fig. 5. Deep neural network model architecture for sensor fault diagnosis

사용되었고 1,000개는 검증데이터 그리고 나머지 1,000개는 테스트를 위해 사용되었다. 사용된 데이터는 그림 1에서의 여러 가지 센서 고장 데이터 유형들로부터 고장 발생 지점을 무작위로 발생시킨 데이터를 사용하였다. 앞서 설명한 데이터를 사용하여 제안한 딥 뉴럴 네트워크를 학습 및 테스트하였다. 사용된 언어는 Python 이며 Keras 라이브러리를 사용하여 라즈베리 파이에서 구현하였다. 센서 고장 진단 성능 비교를 위해 참고문헌<sup>[4]</sup>의 CNN모델을 사용하였다. 모델 파라미터 저장 결과 기존 CNN 모델은 5,695KB이고 inverted residual을 기반으로 설계한 모델의 메모리 용량은 1,442KB로 약 1/4 정도의 메모리 사용량을 줄였다. 이 모델을 라즈베리 파이 위에서 구동시키고 라즈베리 파이에서 판별한 결과와 수집된 데이터는 데이터베이스에 저장하여 조회할 수 있게 MySQL을 통해 구축하였다.

표 2와 그림 6은 CNN 모델과 제안한 네트워크에 의한 분류 정확도를 보여주고 있다. 표와 그래프에서 model1과 model2는 각각 CNN모델과 제안된 모델을 의미한다. 직접적인 정확도 비교를 위해 제안한 네트워크 출력 40개 중에 하나라도 고장이라고 판별할 경우 입력 데이터에 대해 고장이라고 판별하고 전부 normal일 때만 normal이라고 판별하였다. 표와 그래프에서 보는 것과 같이 normal과 erratic의 경우, 이전의 CNN 모델이 정확도가 0.5% 높고 hard-over의 경우 동일하다. 하지만 spike와 stuck의 경우 제안한 분류 모델이 앞서는데, stuck의 경우 이전 모델보다 6%가 높다. 따라서 전체적인 분류 정확도는 CNN과 제안한 모델 각각 97.8%, 98.9%로 이전 모델보다 분류 정확도는 증가한 것을 확인할 수 있다. 참고문헌<sup>[4]</sup>에서 발생지점이 0일 경우, CNN이 stuck에 대해 100%확률로 판별하였으나 무작위 지점에서는 정확도가 낮았다. 이는 normal 신호와 stuck 신호가 동시에 발생할 경우 구분을 잘 못하는 것을 의미한다. 하지만 본 논문에서 설계한 모델은 신호를 구간 분할하여 판별한다. 따라서 normal과 stuck의 특징들을 분리하여 분류가능하기 때문에 정확도를 높이고 신뢰성 높은 분류가 가능함을 알 수 있다.

표 2. CNN 모델과 제안한 딥 뉴럴 네트워크 정확도(%)  
 Table 2. CNN model and proposed deep neural network accuracy(%)

	Normal	Erratic	Hardover	Spike	Stuck
Model1	100	99.5	100	98	91.5
Model2	99.5	99	100	98.5	97.5

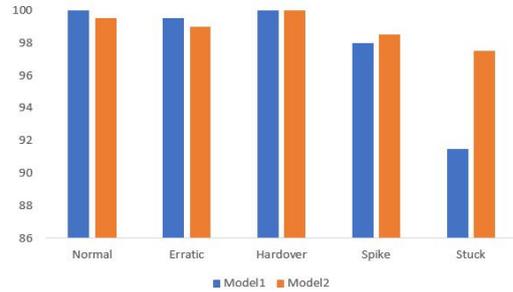


그림 6. 딥 뉴럴 네트워크 모델에 따른 정확도 비교  
 Fig. 6. Accuracy comparison according to deep neural network model

본 논문에서 제안한 딥 뉴럴 네트워크는 출력이 40개이다. 표 2에서의 결과는 1,000개로 이루어진 샘플 데이터 한 개의 입력이 출력 1개로 하나의 데이터 클래스에 대해 판별한다면 그림 5와 같이 구성된 모델의 출력은 40개로 25개의 데이터 요소들에 의한 결과이다. 따라서 입력에 대한 결과는 최근 40초 동안의 센서 고장 판별 결과라고 볼 수 있다. 그림 7은 본 논문에서 제안하는 클래스 판별 방식을 보여주고 있다. 25 포인트 간격으로 클래스에 대해 각 라벨링하였다. Normal, erratic, hardover, spike, stuck은 0부터 4까지 숫자로 라벨링하였다. 그림 7에서는 normal이 0 그리고 hardover가 2로 라벨링되어 있는 것을 나타낸다. 기존의 CNN보다 위의 방법으로 라벨링을 하게 되면 구간 별로 클래스를 판별할 수 있어 고장 발생에 유연하게 대처할 수 있다. 예를 들어, 그림 7의 경우와 같이 노이즈나 다른 환경요인으로 인해 고장이 일시적으로 발생하고 정상상태로 돌아올 경우, CNN에서는 40초가 다 지나갈 동안 계속해서 고장이라고 판별할 것이다. 하지만 본 논문에서 제안한

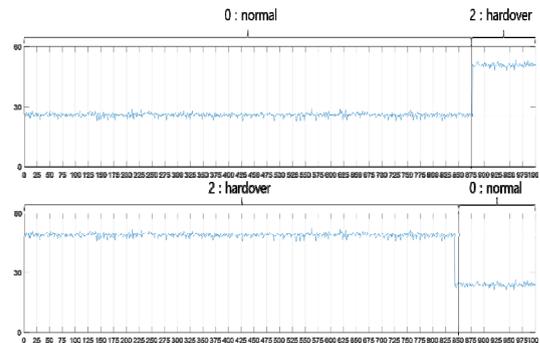


그림 7. 구간 별로 라벨링된 hardover fault 샘플  
 Fig. 7. Hardover fault samples

표 3. 제안한 모델의 센서 고장 분류 혼동행렬(%)

Table 3. Sensor fault classification confusion matrix of the proposed model

	Normal	Erratic	Hardover	Spike	Stuck
Normal	98.55	0.38	0.03	0.68	0.36
Erratic	0.48	99.31	0	0.21	0
Hardover	0	0	100	0	0
Spike	2.92	0	0	97.08	0
Stuck	0.82	0	0	0	99.18

모델은 구간 별로 신호를 받기 때문에 일시적으로 고장으로 판별 후에 계속해서 입력되는 데이터에 대해서는 정상상태라고 판별할 수 있다.

구간 별로 라벨링을 한 뒤 학습을 시켜 테스트한 결과를 표 3에서 혼동행렬로 나타내었다. 전체적 평균 정확도는 98.72%이며, hardover에 대한 정확도가 가장 높다. Spike의 경우 97.08%로 가장 정확도가 낮았으며, 틀린 경우 모두 normal로 판별하였다. Erratic, stuck fault의 경우에도 마찬가지로 normal의 경우로 판별하는 경우가 많은데, 이는 모든 센서 고장은 normal signal로부터 발생하여 그 특징을 가지고 있으므로 normal signal과의 구분이 어려운 것을 알 수 있다.

## VII. 결 론

본 논문에서는 효율적인 센서 고장 검출 및 분류를 위해 inverted residual 구조를 사용한 딥 뉴럴 네트워크를 제안하였다. 대표적인 센서 고장 유형인 erratic, hardover, spike, stuck이 사용되었으며, 전체 센서 데이터의 60%를 학습에 사용하였고 나머지 40% 중에서 각각 20%는 검증용, 테스트용으로 사용하였다. 실험결과 전체적인 분류 성능은 98.9%로 기존 CNN모델에 비해 상승하였으며, 메모리 사용량은 CNN의 약 4분의 1로 메모리 효율을 높였다. 또한 구간별로 분류한 결과 전체 정확도는 98.72%이며, 짧은 구간에 대한 판별이 가능하므로 제안한 방식이 실시간 모니터링에 효과적임을 확인할 수 있었다.

## References

- [1] Gyu-Hyung Lee, Young-Doo Lee, In-Soo Koo, "An RNN-based Fault Detection Scheme for Digital Sensor", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 19, no. 1, pp. 29-35, 2019.  
DOI : 10.7236/JIIBC.2019.19.1.29
- [2] Byeoung Min Mun, Munwon Lim, Seong-Joon Kim, Suk Joo Bae, "Fault Detection and Diagnosis of Smart Factory Equipments Using Wavelet Spectrum", International Journal of Reliability and Applications, Vol. 19, no. 1, pp. 22-30, 2019.  
DOI : 10.33162/JAR.2019.03.19.1.22
- [3] Hyeop-Geon Lee, Young-Woon Kim, Ki-Young, Kim and Jong-Seok Choi, "Design of GlusterFS Based Big Data Distributed Processing System in Smart Factory", Korea Information Electron Communication Technology, pp. 70-75, 2018.  
DOI: <http://dx.doi.org/10.17661/ikiect.2018.11.1.70>
- [4] Jae-Wan Yang, Young-Doo Lee, In-Soo Koo, "Sensor Fault Detection Scheme based on Deep Learning and Support Vector Machine", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 18, No. 2, pp. 185-195, 2018.  
DOI: <http://doi.org/10.7236/JIIBC.2018.18.2.185>
- [5] Sana Ullah Jan, Young Doo Lee, Jungpil Shin and Insoo Koo, "Sensor Fault Classification Based on Support Vector Machine and Statistical Time-Domain Features", IEEE Access, Vol. 5, pp. 8682-8690, 2017.  
DOI: <https://doi.org/10.1109/ACCESS.2017.2705644>
- [6] T. W. Rauber, F. De Assis Boldt, and F.M. Varejão, "Heterogeneous feature models and feature selection applied to bearing fault diagnosis", IEEE Trans. Ind. Electron., vol. 62, no. 1, pp. 637-646, 2015.  
DOI: <https://doi.org/10.1109/TIE.2014.2327589>
- [7] Seung-Jae Kim, Jung-Jae Lee, "A Study on Face Recognition using Support Vector Machine", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 16, No. 6, pp. 183-190, Jun. 2016.  
DOI: <https://doi.org/10.7236/JIIBC.2016.16.6.183>
- [8] Beomsan Kim and Ik Hyun Lee. "Retinal Blood Vessel Segmentation using Deep Learning", Journal of Korean Institute of Information Technology, vol. 17, no.5, pp. 77 - 83, 2019.  
DOI: <https://10.14801/ikiit.2019.17.5.77>
- [9] Seok-Woo Jang, Gyungju Lee, Myunghee Jung. "Effective Detection of Target Region Using a Machine Learning Algorithm", Journal of the Korea Academia-Industrial, vol. 19, no.5, pp. 697 - 704, 2018.  
DOI: <https://10.5762/KAIS.2018.19.5.697>
- [10] Min-Young Kyoung and Hyun-Bean Yi, "A Deep Learning-Based Document Title Detection for Automatic Document Type Classification", Journal of The Institute of Electronics and Information Engineers, vol. 55, no. 9, pp. 53 - 61, 2018.  
DOI: <https://doi.org/10.5573/iee.2018.55.9.53>
- [11] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke and Andrew Rabinovich, "Going Deeper with Convolutions", IEEE Conference

on Computer Vision and Pattern Recognition(CVPR), pp. 1-9, 2015

DOI: <https://doi.org/10.1109/CVPR.2015.7298594>

- [12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition", The IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp. 770-778, 2016.  
DOI: <https://doi.org/10.1109/CVPR.2016.90>
- [13] Karen Simonyan and Andrew Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", International Conference on Learning Representations, arXiv preprint arXiv:1409.1556, 2014.
- [14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. "Mobilenets: Efficient convolutional neural networks for mobile vision applications", arXiv:1704.04861, 2017
- [15] Francois Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 16, No. 6, pp. 183-190, Jun. 2016.  
DOI: <https://doi.org/10.1109/cvpr.2017.195>
- [16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. CVPR, arXiv:1801.0438, 2018.  
DOI: <https://doi.org/10.1109/cvpr.2018.00474>

#### 구 인 수(정회원)



- 1996년: 건국대학교 전자공학과 졸업 (학사)
- 1998년: 광주과학기술원 정보통신공학과 졸업 (석사)
- 2002년: 광주과학기술원 정보통신공학과 졸업 (박사)
- 2002년~2004년: 광주과학기술원 연구교수
- 2003년~2004년: 스웨덴왕립공과대학 박사 후 연수과정
- 2005년~현재: 울산대학교 전기공학부 교수
- 관심분야: 차세대 통신시스템, 무선센서 네트워크

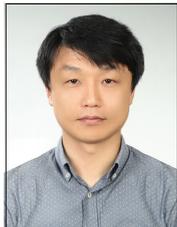
#### 저 자 소 개

##### 양 재 완(준회원)



- 2017년: 울산대학교 전기공학부 (학사)
- 2017년~2019년: 울산대학교 전기공학부 석사졸업
- 2019년~현재: (주)길온 연구원
- 관심분야: 신호처리, 인공지능, 인지무선 시스템

##### 이 영 두(정회원)



- 2007년: 울산대학교 전기전자정보시스템 공학부 (학사)
  - 2009년: 울산대학교 전기전자정보시스템 공학부 (석사)
  - 2013년: 울산대학교 전기전자정보시스템 공학부 (박사)
  - 2013년~현재: 울산대학교 전기공학부 연구교수
- 관심분야: 신호처리, 인공지능 기반 네트워크, 무선인지 네트워크, 수중센서 네트워크, 차세대 통신시스템