| **TinyML을 통한 소리 인식 및 위치 탐지**  **요 약**  지진 등의 사고 현장 같은 불안정한 장소에서 사고 피해자들을 찾기 위한 AI 모델링을 구현한다. 드론에 장착하기 위해 가볍고 저전력으로 사용가능한 TinyML 모델을 학습시켜 드론에 장착하여 사고 피해자들의 소리를 감지하고 위치를 탐지하는 시스템을 구현한다. Sound Classification과 Sound Localization을 혼합하여 Edge Impulse 플랫폼을 사용하여 모델을 학습을 시도한다. 사용할 데이터는 기존에 존재하는 데이터셋 및 직접 녹음한 데이터를 통해 학습을 진행한다. 학습한 모델을 nano 33 ble에 배포하고 드론에 장착시켜 소리를 인식하고 위치를 탐지시킨다. |
| --- |

# 1. 서론

## 1.1. 연구배경

2023년 3월 튀르키예 대지진으로 8만명 이상의 사람이 목숨을 잃었다. 이러한 재난 상황에서 생존자를 빠르게 구조하는 것은 매우 중요하다. 기존의 드론 기술은 이미 재난 구조 작업에 활용되고 있지만, 생존자의 위치를 정확하게 파악하는 것이 여전히 어려운 문제가 있다. 기존의 드론 기술은 대부분 카메라나 적외선 카메라를 이용하여 생존자를 탐지하는데, 적외선 카메라의 경우 높은 가격과 복잡한 사용법 때문에 일반적인 상황에서 사용하기 어려운 단점이 있다. 또한, 카메라로만 생존자를 탐지하는 경우 생존자의 음성이나 소리를 감지하는 것이 어렵기 때문에, 이러한 상황에서는 생존자의 위치를 정확하게 파악하기 어려운 경우가 발생할 수 있다.

따라서, 이러한 문제점을 TinyML 기술을 활용하여 해결하고자 한다. TinyML 기술은 저전력, 저비용, 소형 디바이스에 적합한 머신 러닝 기술로, 드론과 같은 소형 디바이스에서도 사용이 가능하다. 이를 활용하여 드론이 생존자의 음성이나 소리를 감지하여 위치를 파악하고, 이를 기반으로 생존자를 신속하게 구조할 수 있는 시스템을 개발할 것이다. 이러한 시스템은 재난 구조 작업에서 매우 유용하게 활용될 수 있으며, 이에 따라 많은 인명 구조 작업에서 적극적으로 활용될 것으로 기대된다.

## 1.2. 연구목표

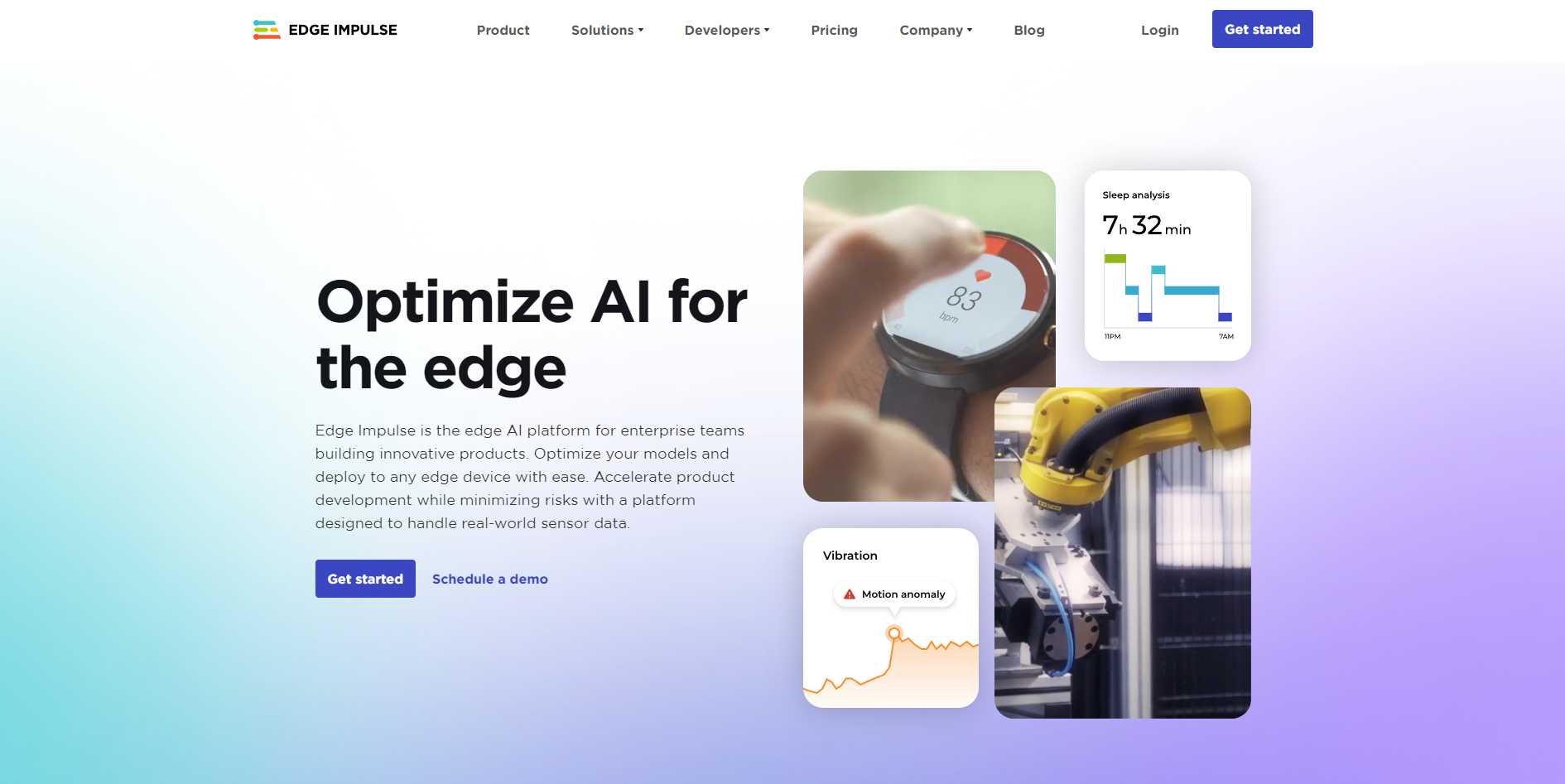
본 연구의 목표는 Edge Impulse 플랫폼을 활용하여 TinyML 모델을 학습시켜, 드론에 장착하여 사고 피해자들의 소리를 감지하고 위치를 탐지하는 시스템을 구현하는 것이다. 드론에 장착하여 사용하기 위해서는 가볍고 적은 자원을 사용하면서도 정확한 기능 수행이 요구 된다. 따라서 이런 조건을 만족하기 위해 TinyML을 사용하여 이를 수행한다. TinyML을 통해 재난 구조 작업에서 드론이 보다 정확하고 신속하게 생존자를 찾을 수 있으며, 재난 구조 작업의 성공 확률을 높일 수 있도록 하는 것이 이번 연구의 핵심 목표이다.

# 2. 배경 이론

## 2.1. TinyML

TinyML은 작고 저전력인 인공지능 모델을 만들고 배포하는 기술로, 임베디드 시스템에서도 활용할 수 있다. 이를 통해 기기 내부에서 지능적인 의사 결정을 수행하고, 외부 서버와의 통신을 최소화하여 보안성을 높이고 전력 소비를 줄일 수 있다. TinyML 기술은 IoT, 스마트 홈, 건강 모니터링 등 다양한 분야에서 활용될 수 있으며, 본 프로젝트에는 드론의 임베디드 시스템으로 사용한다. 이를 통해 작고 강력한 인공지능 모델을 개발하고 배포함으로써, 보다 효율적이고 안전한 임베디드 시스템을 구현할 수 있다.

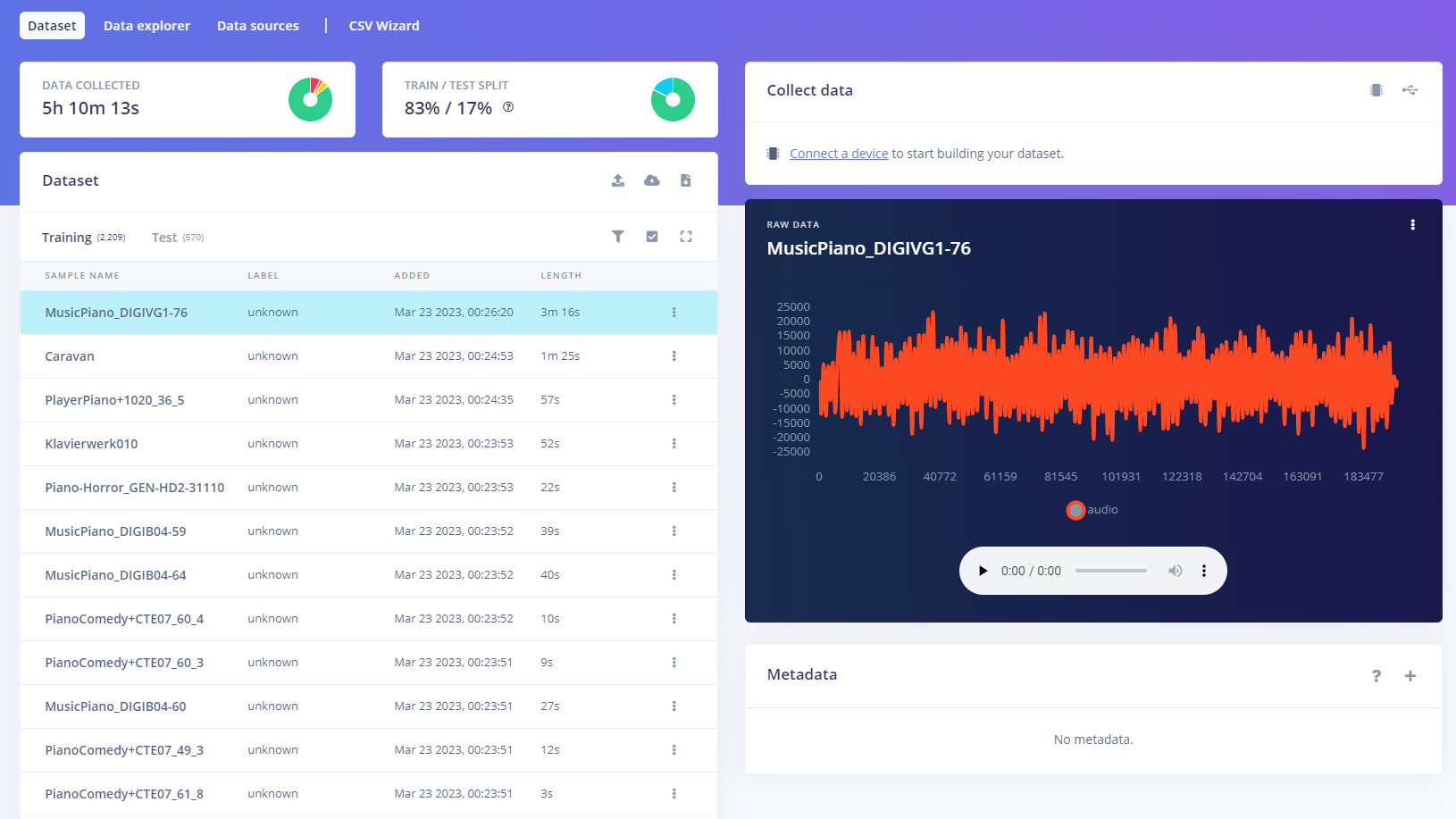
## 2.2. Edge Impulse



<그림 1 Edge Impulse 1>

Edge Impulse는 작고 효율적인 TinyML 모델을 만들기 위한 종합적인 개발 플랫폼이다 [1]. 사용자는 Edge Impulse 웹 사이트에서 데이터 수집부터 모델 학습까지 전 과정을 진행할 수 있으며, 학습시킨 모델을 다양한 디바이스와 통합이 가능하다. 또한, Edge Impulse를 통해 데이터 전처리, 모델 최적화, 배포 등의 과정을 자동화하여, 비전문가도 쉽게 인공지능 모델을 개발하고 배포할 수 있다. 이를 통해, IoT, 스마트 홈, 로봇, 드론 등 다양한 임베디드 시스템에서 강력하고 효율적인 인공지능 모델을 쉽게 개발할 수 있다.

### 2.2.1. 데이터 수집



<그림 2 Edge Impulse 2>

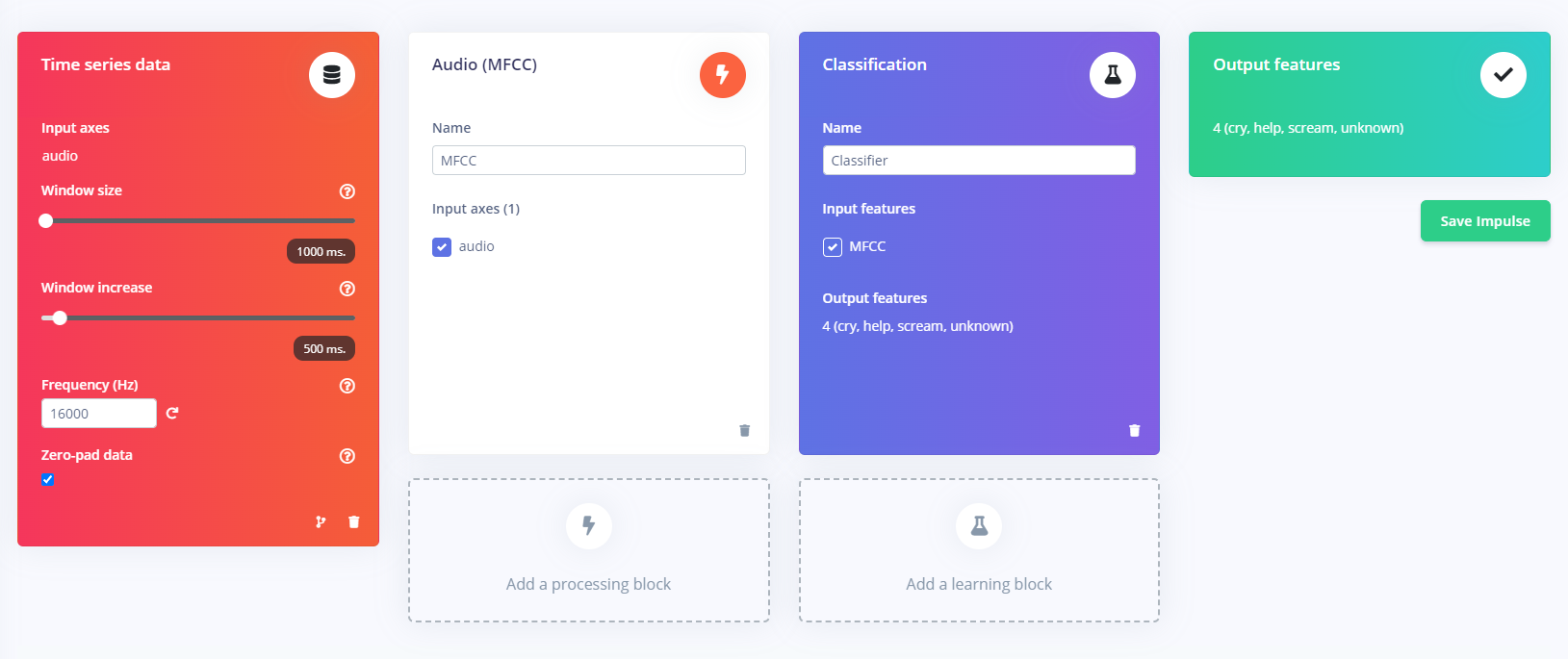
데이터는 사용 용도에 따라 이미지, 음성, 동작 데이터등을 사용할 수 있다. 데이터 수집에는 다양한 방식이 있다. 기존의 파일 혹은 CSV 등의 형태의 파일을 Edge Impulse에 업로드 하여 데이터를 얻을 수 있다. 또는 Edge Impulse와 연동한 마이크, 스마트폰, 아두이노 등의 디바이스를 통해 직접 녹음 및 촬영을 통해 이미지, 음성, 동작 데이터를 얻을 수 있다.

각 데이터는 라벨링을 하여 분류 할 수 있으며, train과 test 데이터로 데이터를 나눌 수 있다. 이를 통해 학습할 때 올바른 분류를 수행하고 정확도를 높일 수 있다. 음성 데이터의 경우 원하는 길이 만큼 split하여 원하는 부분만을 추출할 수 있다.

데이터 수집은 Edge Impulse에서 다양한 형태의 데이터를 확보하기 위한 과정이다. 데이터 수집 단계에서는 충분히 다양하고 대표적인 데이터를 수집하고, 데이터의 품질을 유지하는 것이 중요하다. 이는 모델의 성능과 신뢰도에 직접적인 영향을 미치게 된다.

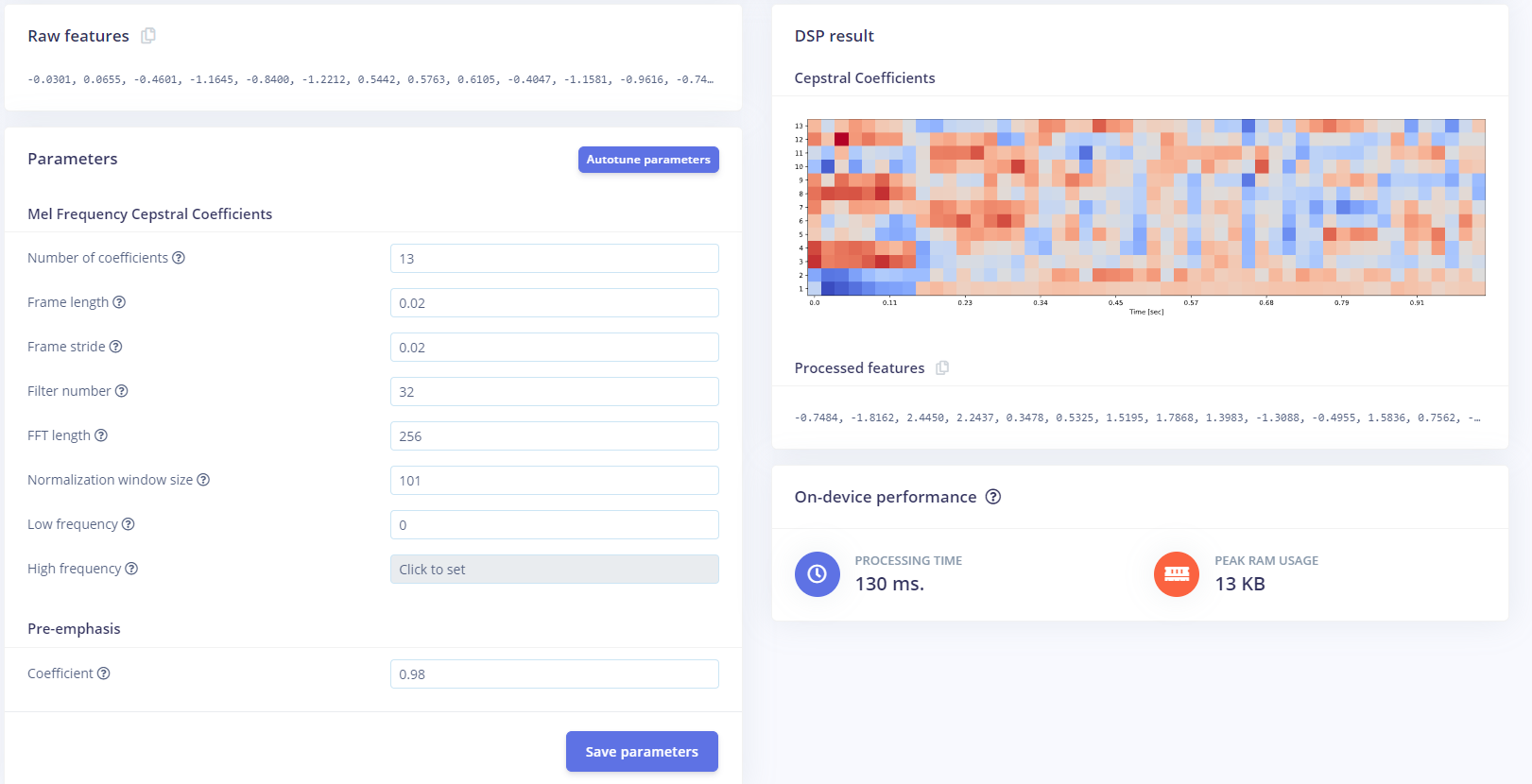
### 2.2.2. Impulse Design

Impulse Design은 Edge Impulse에서 데이터 전처리 및 모델 구성을 위한 핵심적인 단계이다. 데이터는 종종 노이즈, 변형, 불균형 등의 문제를 가지고 있을 수 있다. 이러한 데이터를 직접 모델에 적용하기 전에 적절한 전처리 과정을 거쳐 데이터의 품질을 향상시키고 모델의 성능을 최적화할 수 있다.



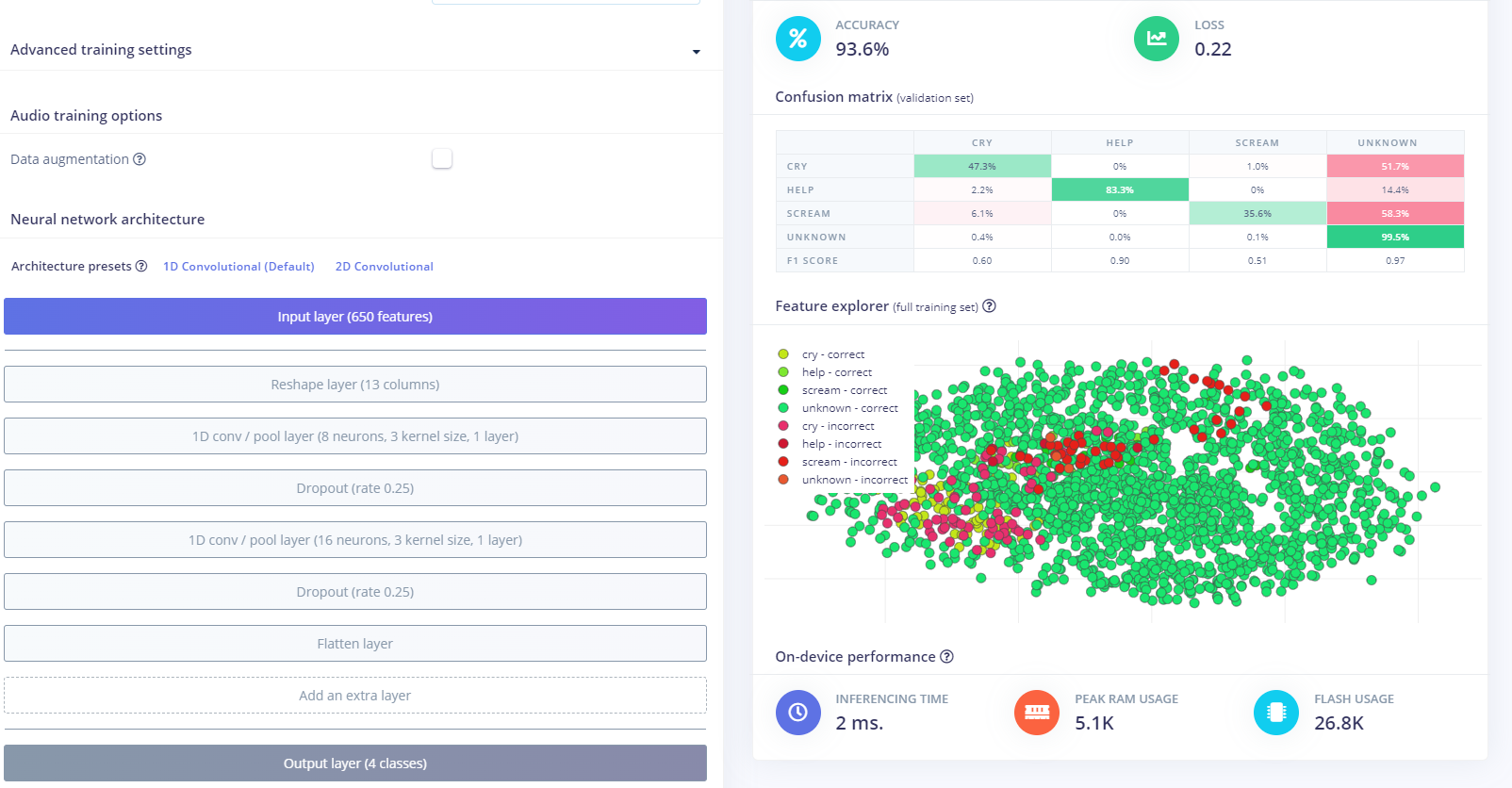
<그림 3 Edge Impulse Impulse Design>

수집한 데이터는 Impulse Design을 통해 전처리 후 모델링을 할 수 있다. 음성 데이터를 전처리하는 다양한 Processing Block(MFCC, MFE, Spectogram) 등이 있다. 추출한 데이터를 Learning Block(Classification, Regression, Key-word Spotting) 등의 모델을 거쳐 Output을 내는 모델을 구현할 수 있다.



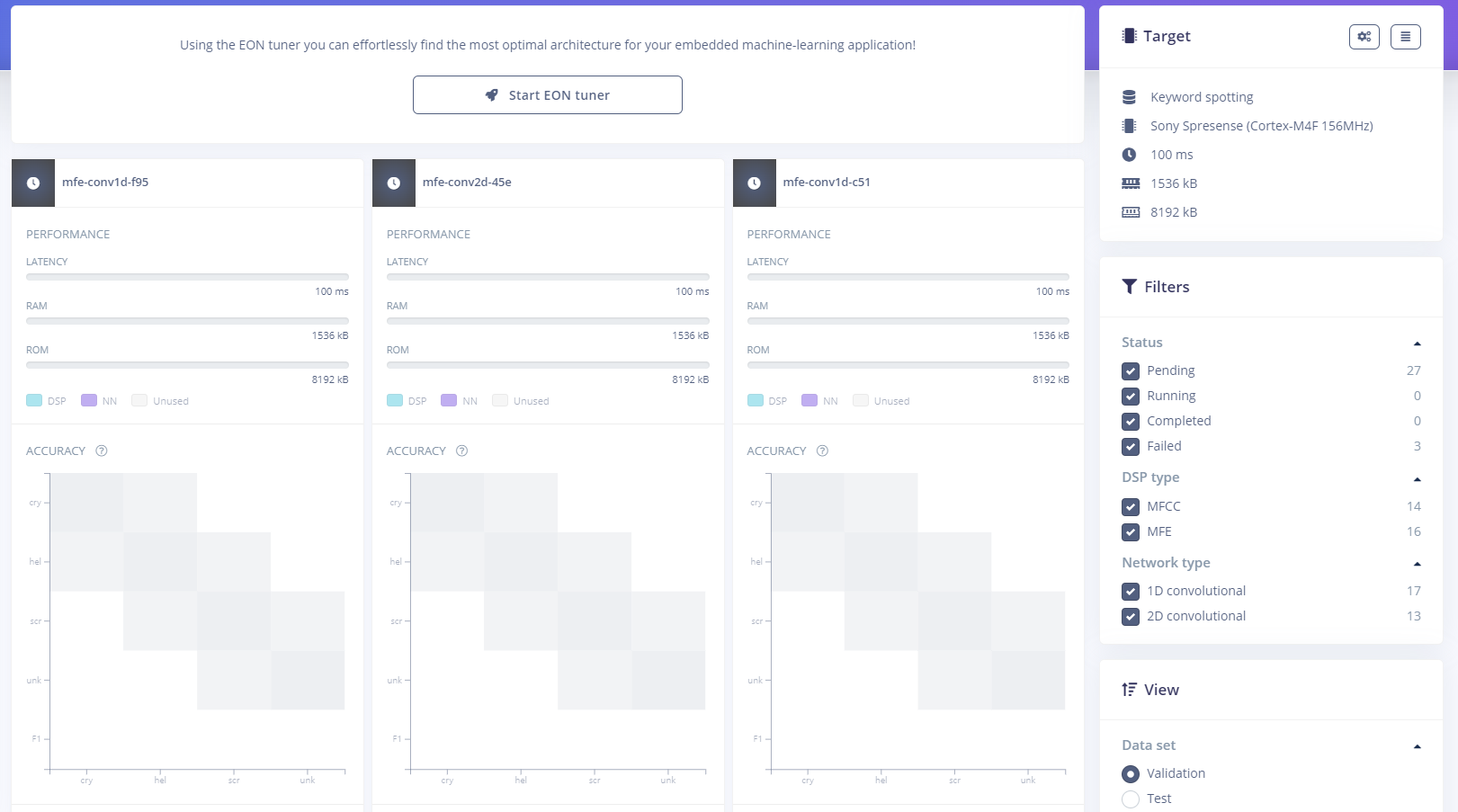
<그림 4 Edge Impulse Processing Block>

데이터 전처리는 수집한 데이터의 feature을 추출한다. Impulse Design에서는 다양한 Processing Block을 활용하여 데이터를 전처리한다. 예를 들어 음성 데이터의 경우 MFCC (Mel Frequency Cepstral Coefficients), MFE (Mel Filterbank Energies), Spectrogram 등의 처리 블록을 사용하여 주파수, 에너지, 스펙트럼 등의 특징을 추출할 수 있다. 이러한 전처리 과정을 통해 데이터의 중요한 특성을 강조하고 잡음이나 불필요한 정보를 제거하여 모델에 더 의미 있는 데이터를 제공한다. 추출한 정보를 Learning Block으로 보낸다.



<그림 5 Edge Impulse Learning Block>

Processing Block이 전달한 정보를 Learning Block을 통해 학습을 진행한다. Learning Block을 선택하여 모델링을 진행하며, 분류(Classification), 회귀(Regression), 키워드 감지(Key-word Spotting) 등의 다양한 학습 모델을 적용할 수 있다. Impulse Design 단계에서는 모델의 구조, 레이어 수, 하이퍼파라미터 등을 조정하여 최적의 모델을 구성할 수 있다. 학습 후에는 정확도 및 Loss값의 추정치와 Device에서 작동 시의 사양값의 추정치를 얻을 수 있다.

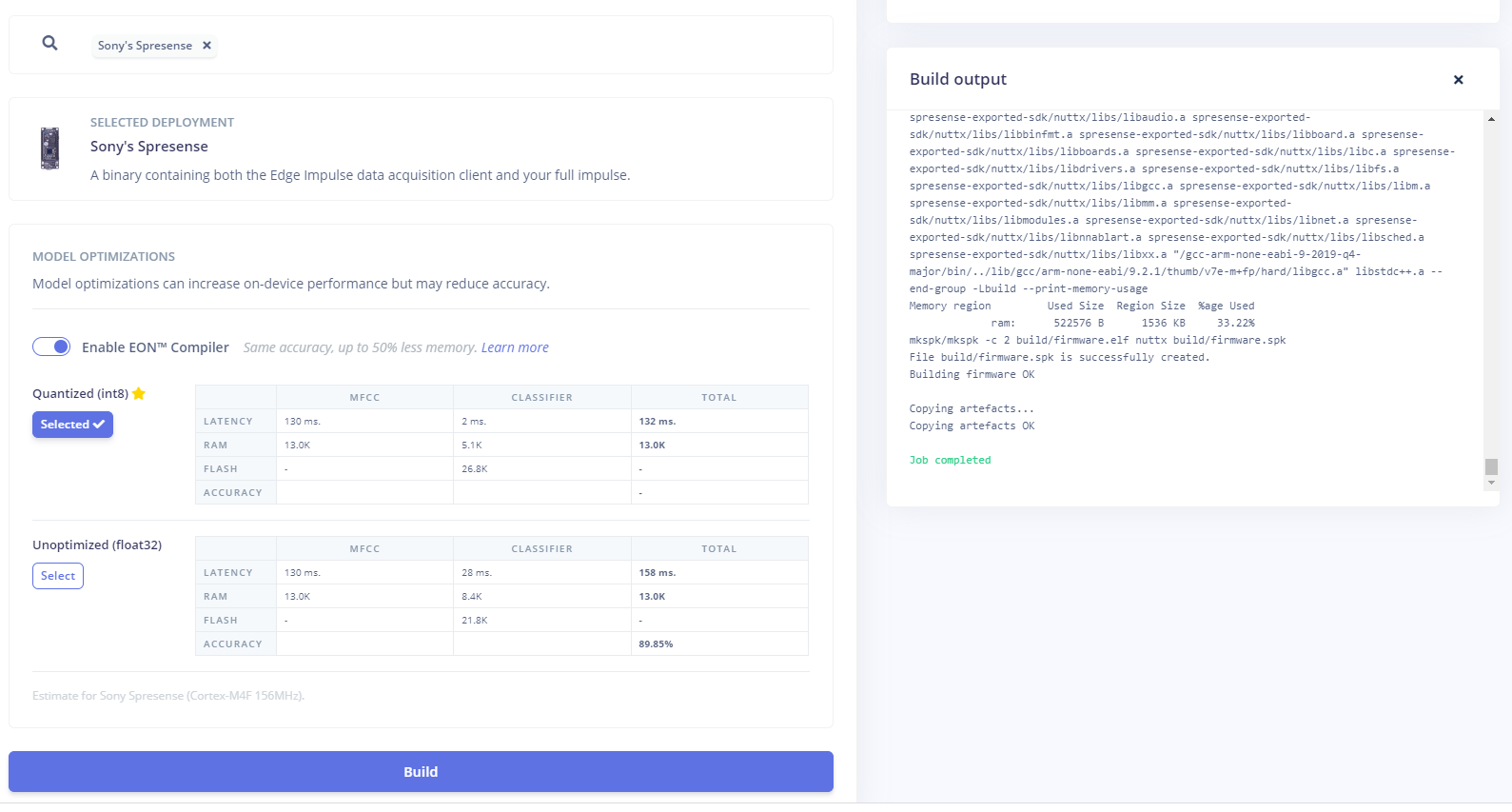


<그림 6 Edge Impulse EON Tuner>

또한 Edge Impulse에서는 EON Tuner라는 기능을 지원한다. 모델의 학습 과정을 제어하고 모델의 구조, 학습 속도, 정규화 등과 관련된 매개 변수를 하이퍼파라미터라고 한다. EON Tuner는 자동화된 하이퍼파라미터 최적화를 수행한다. 사용자가 지정한 하이퍼파라미터 공간에서 여러 가지 조합의 모델을 사용하며 모델에 사용할 디바이스 및 데이터의 형태를 고려한다. 여러 조합에서 모델의 정확도, 학습 시간, 메모리 사용량 등과 같은 성능 지표를 고려하여 하이퍼파라미터 최적화를 수행하고 모델의 성능을 향상 시킬 수 있다.

결론적으로, Impulse Design은 데이터의 전처리와 모델의 구성에 대한 전략적인 계획을 수립하는 단계로서 중요성을 갖는다. 올바른 전처리 기법과 모델의 선택은 모델의 성능을 향상시키고 데이터의 특징을 잘 반영할 수 있게 해준다. 이를 통해 실제 응용 분야에서 정확하고 신뢰할 수 있는 결과를 얻을 수 있다. 따라서 Impulse Design 단계에서 충분한 시간과 노력을 투자하여 데이터의 전처리와 모델의 구성을 신중하게 결정하는 것이 중요하다.

### 2.2.3. Deployment



<그림 7 Edge Impulse Deployment>

모델이 학습되었다면 그 모델을 Edge Impulse에서 배포할 수 있다. 배포에는 다양한 방식이 있는데 C++ library나 Arduino Library 등 여러 IDE에서 실행하여 배포 가능한 파일을 얻을 수 있다. 또는 Edge Impulse에서 지원하는 각종 실행 파일을 받아 연결하여 바로 TinyML을 구현할 수 있다. 마지막으로 QR코드를 통해 스마트폰 등에서도 모델을 확인할 수 있다.

## 2.3. Sound Processing

### 2.3.1. Multi Channel Sound

멀티 채널(Multi Channel)은 여러 개의 마이크로폰이나 스피커를 사용하여 다중 채널의 음향 신호를 처리하는 기술이다. 이를 통해 더욱 정확하고 상세한 오디오 정보를 추출하고 분석할 수 있다. 멀티 채널 기술은 음향 분야에서 다양한 응용 분야에 적용되며, 소리 분류 및 위치 파악에 있어서 특히 다음과 같은 중요한 역할을 한다.

멀티 채널을 사용하면 여러 개의 마이크로폰으로부터 동시에 소리를 수집할 수 있다. 이는 실시간 오디오 처리나 대규모 데이터셋에서의 빠른 처리 속도를 보장한다. 이러한 다중 소리 데이터를 활용하여 소리 분류를 수행한다. 멀티 채널 소리 분류는 다음과 같은 과정을 거친다.

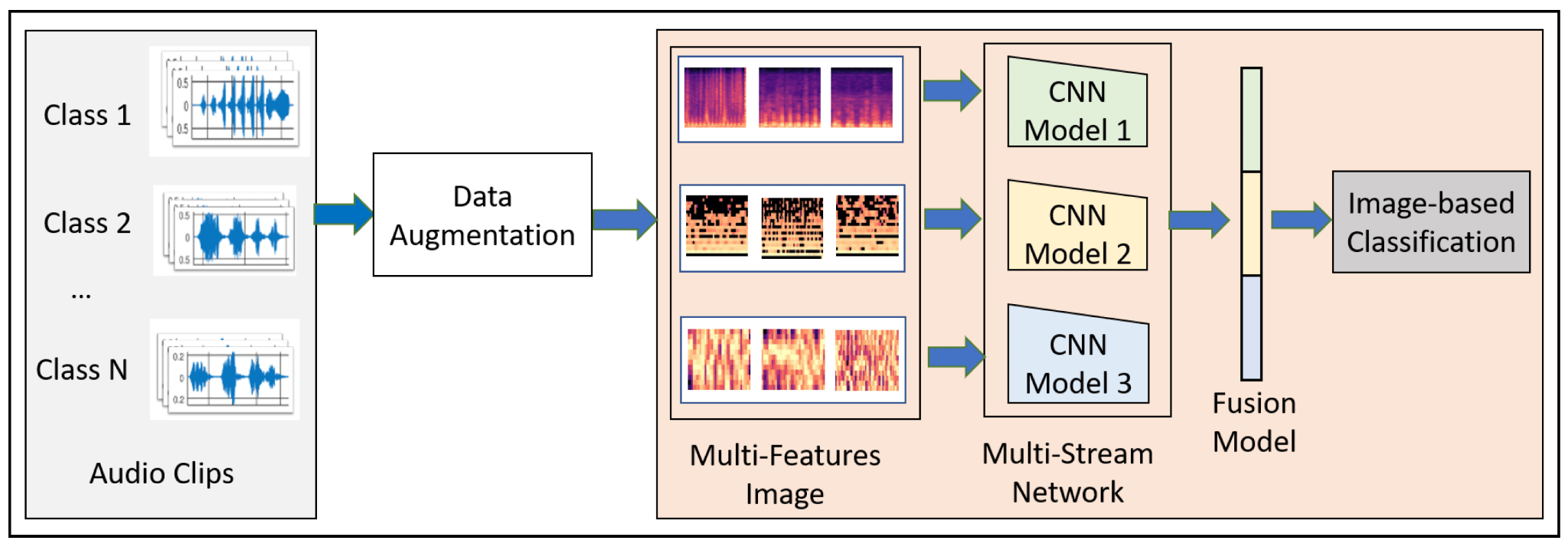
멀티 채널에서 수집한 소리 데이터를 동기화하여 분석한다. 각 채널에서 추출한 소리의 특징(주파수, 진폭, 위상 등)을 계산한다. 다중 채널에서 추출한 소리 특징을 결합하여 소리 패턴을 생성한다. 생성된 소리 패턴을 기반으로 다양한 분류 알고리즘을 적용하여 소리를 분류한다. 멀티 채널 소리 분류를 통해 다중 소리 환경에서 정확한 소리 분류가 가능해지며, 음향 신호 처리와 관련된 다양한 응용 분야에서 활용된다.

멀티 채널을 사용하면 다수의 마이크로폰을 이용하여 소리의 발생 위치를 추정할 수 있다. 멀티 채널 소리 위치 파악은 소리의 도착 시간 차이(Time Difference of Arrival, TDOA)를 계산하여 소리의 발생 위치를 추정하는 과정이다.

멀티 채널 소리 위치 파악을 위해서는 여러 개의 마이크를 공간에 분산하여 배치한다. 일반적으로 마이크는 다양한 방향과 거리에서 소리를 수집할 수 있도록 배치된다. 각 채널의 마이크에서 소리를 수집한다. 이렇게 수집된 오디오 신호는 각 채널의 입출력을 통해 디지털 형태로 처리된다. 각 채널에서 수집된 신호는 정확한 타이밍으로 동기화되어야 한다. 이를 위해 시간 동기화 기술이 사용된다. 동기화된 오디오 신호를 분석하여 소리의 특징을 추출한다. 이를 위해 주파수 분석, 시간-주파수 변환 등의 기법이 사용될 수 있다. 추출된 특징을 기반으로 소리의 위치를 계산한다. TDOA를 계산하여 소리의 방향을 파악하는 TDOA 추정 알고리즘을 사용한다. 이를 통해 소리의 원래 방향과 위치를 추정한다.

### 2.3.2. Sound Classification

Sound classification은 소리의 특징을 분석하여 해당 소리의 종류를 분류하는 인공지능 기술 중 하나다 [2, 3]. 관련 연구에서는 CNN(Convolutional Neural Network) 과 TDSN(Tensor Deep Stacking Network)을 훈련하기 위해 환경 소리의 스펙트로그램 이미지를 사용하여 Sound Classification을 구현했다.



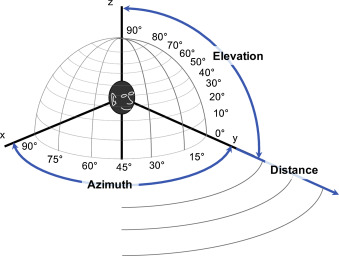
<그림 8 Sound Classification>

Sound classification을 통해, 본 프로젝트에서는 사고 현장에서 살려 달라는 소리나 울음 소리 등을 인식할 수 있게 모델을 학습시킨다. 이를 구현하기 위해서는, 먼저 해당 소리 데이터를 수집하고, 이를 토대로 모델을 학습시켜야 한다. 학습된 모델은 드론 등의 임베디드 시스템에 적용되어, 실시간으로 소리를 분류하고, 사고 피해자들의 상황 파악 및 구조 작업에 활용할 수 있다. 이를 통해, 재난 현장에서 빠른 대응과 효과적인 구조 작업을 수행할 수 있게 된다.

### 

### 2.3.3. Sound Localization

Sound localization[4]은 소리가 발생한 위치를 인식하는 인공지능 기술 중 하나다. 관련 연구에서는 CDR(coherent-to-diffuse power ratio)의 추정치를 기반으로 한 ASN(acoustic sensor network)에서 음원 위치 파악을 조사한다. 실제 음향과 마이크 쌍 사이의 거리를 통해 위치를 예측한다. 획득된 거리 추정치는 음원의 위치를 ​​추정하기 위해 알려진 상대 노드 위치를 갖는 ASN에 의해 ​​사용된다.



<그림 9 Sound Localization>

Sound Localization을 효율적으로 하기 위해서는 음성 데이터가 Azimuth, Elevation, Distance 등의 값을 가지고 있어야 한다. 하나의 음성 데이터에 3개의 라벨을 라벨링하여 데이터를 만들고 Regression을 통해 학습시킨다.

Regression을 사용하여 학습된 모델은 소리의 특성을 기반으로 해당 소리가 발생한 위치를 예측할 수 있다. Sound localization은 음성 인식, 화상 회의 등의 다양한 분야에서 활용될 수 있으며, 이를 위해 다양한 학습 방법이 사용된다. 본 프로젝트에서는, 드론의 임베디드 시스템이 소리를 인식하고, 그 소리가 발생한 위치를 파악하여, 구조 작업에 적극 활용한다.

Regression은 sound localization 중 하나로, 학습된 모델은 입력 데이터와 출력 데이터 간의 관계를 파악하여, 새로운 입력 데이터가 주어졌을 때, 해당 출력 데이터를 예측할 수 있다.

### 

### 2.3.4. Integration of Sound Classification & Localization

Sound classification은 소리의 종류를 분류하는 작업에 초점을 두고 있으며, Sound localization은 소리의 발생 위치를 파악하는 작업에 초점을 두고 있다. 그러나 최근에는 이 두 작업을 함께 처리하는 연구가 늘어나고 있다.

예를 들어, Sound classification과 Sound localization을 모두 다루는 연구로는 SeldNet [5], CRNN [6] 등이 있다. 이번 프로젝트에서는 이와 같은 기존 연구들에서 사용된 모델을 참고하여, 드론의 임베디드 시스템에서 Sound classification과 Sound localization을 함께 처리하려고 한다.

이와 같이 기존의 Sound classification과 Sound localization을 혼합하여 사용한 논문 연구를 참고하여, 이번 프로젝트에서는 드론에 이를 적용하여 Sound classification과 Sound localization을 함께 처리할 것이다.

# 3. 프로젝트 내용

## 3.1. 데이터 수집

데이터셋 수집 단계에서는 기존에 수집된 소리 데이터셋을 참고하여 프로젝트에 필요한 소리를 포함하고 있는 데이터셋을 수집하는 것이 중요하다. 이를 위해 공개된 오픈 데이터셋[7]을 사용하거나 직접 녹음하여 데이터셋을 만들었다. Sound Classification을 위해 만든 데이터셋은 다음과 같다.

| **Category** | **Number of Samples** |
| --- | --- |
| cry | 293개 |
| help | 401개 |
| scream | 471개 |
| unknown | 1044개 |

<표 1 데이터셋>

이 데이터셋에서 있는 여러 종류의 소리 중에서 우리가 필요로 하는 살려달라는 소리나 우는 소리 등을 찾아내어 추출한다. 나머지 소리는 unknown으로 라벨링을 한다. 이 과정에서는 데이터셋에서 사용할 소리에 대한 정보와 해당 소리의 레이블링 정보를 모두 수집하여 Edge Impulse에 명시해야 한다.

Sound Localization을 위한 데이터 같은 경우, 찾아본 데이터셋 들의 채널이 8개 이상이었다. 그러나 사용한 nano 33 ble은 기본적으로 2개의 채널을 지원한다. 또한 Edge Impulse에서 음성 데이터를 수집할 때 단일 채널을 사용하였기에 azimuth 값이나 elevation 값을 이용한 학습이 불가능했다.

## 3.2. Edge impulse를 통한 학습

데이터 전처리, 모델 학습, 모델 평가 단계는 Edge Impulse를 통해 진행한다.

첫 번째로 데이터 전처리 단계에서는 수집된 데이터셋에서 노이즈 제거 및 스케일링, 필터링, 주파수 변환 등의 전처리 과정이 진행된다. 전처리를 위한 Processing Block은 MFCC를 사용한다. 추출한 features들을 Learning Block으로 보낸다.

두 번째로 모델 학습 단계에서는 Edge Impulse에서 제공하는 다양한 알고리즘을 이용하여 소리 분류 모델을 학습한다. 사용한 Learning Block은 Classifier이며 Learning Rate는 기본 값인 0.005를 통해 Output Layer는 4개를 도출한다.

세 번째로 모델 평가 단계에서는 학습된 모델을 테스트 데이터셋으로 검증하여 모델의 성능을 평가한. 이 과정에서 정확도, 정밀도, 재현율, F1-score 등의 평가 지표를 사용하여 모델의 성능을 분석한다. 이를 통해 모델의 성능을 개선하고 최종 모델을 선정한다.

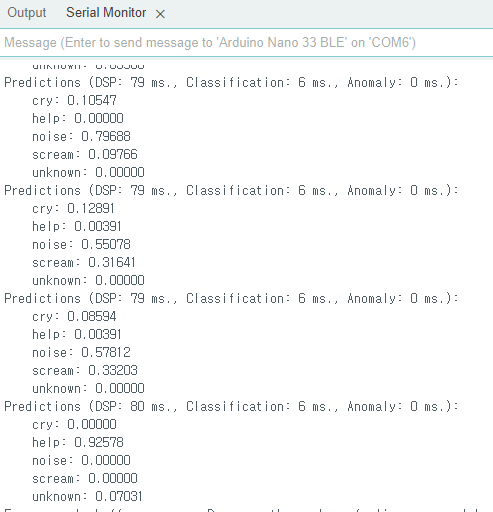
## 3.3. Device 설계

프로젝트에 임베디드 시스템으로 학습한 모델을 넣는 device로는 nano 33 ble[9]을 사용한다. Nano 33 ble은 기본적으로 2대의 마이크로폰이 장착되어 있으며 이를 통해 소리를 인식하고, RGB led를 통해 자신이 소리를 인식했음을 시각적으로 표시할 수 있다.

## 3.4. Device에 Deploy

학습한 모델을 export하여 디바이스에서 실행할 수 있는 형태로 변환한다. Nano 33 ble이 extension board가 없을 시 Edge Impulse에서 제공한 파일을 그대로 사용할 수가 없다. 따라서 Arduino IDE에 사용할 수 있는 ino 파일을 export한 뒤 nano 33 ble을 연결한 상태에서 ino 파일을 실행하여 deploy를 진행한다.

## 4.프로젝트 결과



<그림 10 Arduino IDE 결과>

Deployment에서 얻은 ino 파일을 nano 33 ble가 연결된 상태에서 Arduino IDE로 실행한다. 모델을 nano 33 ble에 적용시키고 1초 간격으로 마이크 인식을 통해 소리를 분류한다.



<그림 11 Nano 33 ble 결과>

Nano 33 ble이 판단하여 소리가 cry, help, scream 의 추정 값이 0.8 이상일 때 붉은 LED가 점등하여 소리 인식을 표시한다.

# 5. 결론

## 5.1 기대 효과

이 프로젝트는 딥러닝 기술을 활용하여 드론 사고 발생 시, 사고 피해자들의 위치를 빠르고 정확하게 파악할 수 있는 솔루션을 제공하는 것을 목표로 하고 있다.

데이터셋 수집부터 모델 학습까지 edge computing 환경에서 진행되어 성능과 속도 면에서 우수한 결과를 기대할 수 있다. 또한 디바이스에 deploy하여 드론에 적용함으로써, 사고 발생 시 보다 빠르고 정확한 대응이 가능해지며, 이는 인명 구조에 큰 도움이 될 것이다.

데이터셋의 요구사항을 충족시키기 위해 방향 정보를 포함하여 수집하였으며, Edge Impulse에서는 classification을 모델링하여 효율적인 모델 학습을 할 수 있도록 구성하였다. 또한 디바이스에 대한 요구사항 역시 Edge Impulse와의 호환성과 하드웨어 성능이 충분하도록 설정하여 문제 없이 작동할 수 있도록 구성하였다.

이를 통해 드론 사고 발생 시, 사고 피해자 위치 파악에 대한 대처 능력이 향상되어 빠른 대응이 가능해지며, 이는 인명 구조에 큰 도움이 될 것이다. 이 프로젝트를 통해 딥러닝 기술이 실제 사회에서 활용될 수 있는 예시를 보여주고, 드론 사고 발생 시 대처 능력을 향상시켜 인명 구조 분야에 기여할 수 있는 선구적인 결과물이 될 것이다.

## 5.2 추후 연구 방향

데이터 전처리를 통해 multi channel을 가진 소리 데이터를 만들어 azimuth 값 및 elevation 값을 가진 음성 데이터를 만든다. 만든 데이터로 임베디드 시스템이 Sound Localization을 학습할 수 있는 모델을 만들며 Classification과 동시에 모델을 작동할 수 있게 구축한다. 최종적으로는 드론에 장착하여도 제대로 소리를 인식할 수 있게 정확성을 높이며 다양한 언어의 정보를 데이터로 넣어 더 범용적인 사용이 가능하게끔 설계한다.

# 6. 참고문헌

[1] Edge Impulse, <https://www.edgeimpulse.com/>

[2]A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey and P. Tiwari, "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network," in IEEE Access, vol. 7, pp. 7717-7727, 2019, doi: 10.1109/ACCESS.2018.2888882.

[3] Sajjad Abdoli, Patrick Cardinal, Alessandro Lameiras Koerich, End-to-end environmental sound classification using a 1D convolutional neural network, Expert Systems with Applications, Volume 136, 2019,

Pages 252-263, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2019.06.040.

(<https://www.sciencedirect.com/science/article/pii/S0957417419304403>)

[4] A. Brendel and W. Kellermann, "Distributed Source Localization in Acoustic Sensor Networks Using the Coherent-to-Diffuse Power Ratio," in IEEE Journal of Selected Topics in Signal Processing, vol. 13, no. 1, pp. 61-75, March 2019, doi: 10.1109/JSTSP.2019.2900911.

[5] Sharath Adavanne, Archontis Politis, Joonas Nikunen, Tuomas Virtanen, "Sound Event Localization and Detection of Overlapping Sources Using Convolutional Recurrent Neural Networks" , <https://doi.org/10.48550/arXiv.1807.00129>

[6] Sławomir Kapka, Mateusz Lewandowski, "Sound source detection, localization and classification using consecutive ensemble of CRNN models", <https://doi.org/10.48550/arXiv.1908.00766>

[7]Acoustic Sensing of Worker Accidents,, https://studio.edgeimpulse.com/public/111611/latest

[8] Nano 33 ble, https://store.arduino.cc/products/arduino-nano-33-ble