Tiny ML을 이용한 부품 판별 시스템

담당 지도 교수 : 백윤주

201501199 윤태훈

201724519 이강윤

목차

1. 과제 배경 및 목표

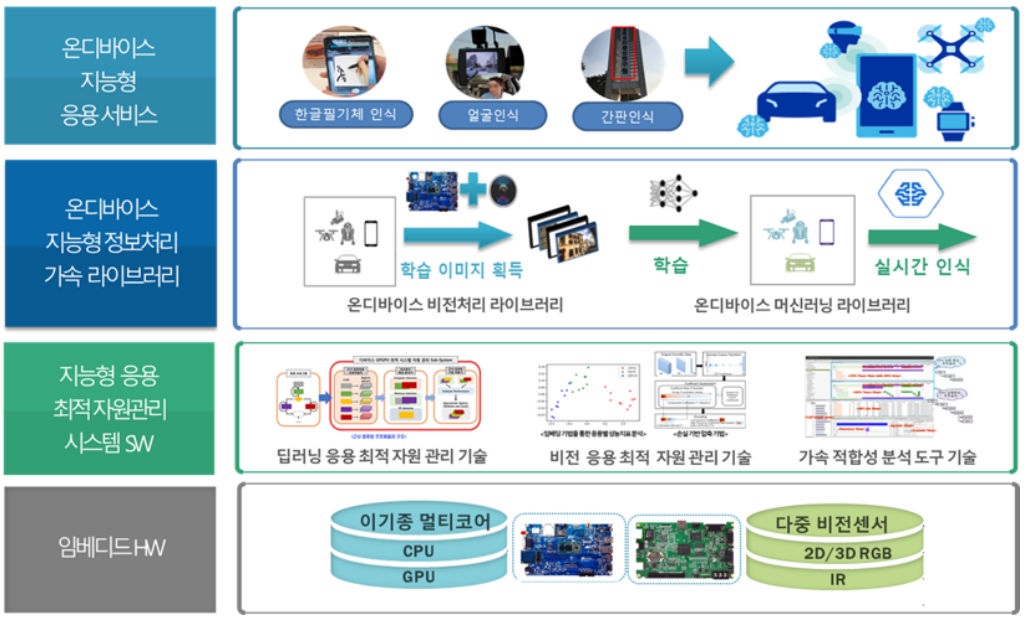
1.1 과제 배경

초소형 임베디드 시스템과 AI 기술의 상호작용은 최근 기술 발전의 한 축으로 주목받고 있다. 특히 "Tiny ML"이라고 불리는 이러한 초소형 AI 시스템은 다양한 산업 분야에서 혁신적인 가능성을 제시하고 있다. AI 기술의 발전으로 객체 탐지와 같은 고급 기능이 이제는 작고 효율적인 임베디드 시스템에서도 실현 가능해졌다.

이러한 발전은 부품의 판정과 같은 분야에서도 접목 시킬 수 있다. 예를 들어, 부품의 분류 과정에서 AI 기반의 객체 탐지 기술을 활용하면, 이전에 수작업으로 이루어지던 작업을 자동화하고 높은 정확도로 원하는 결과를 얻을 수 있다.

최근에는 스마트폰이 지속적으로 개발됨에 따라서 내부를 구성하는 부품 역시 지속적으로 변화되고 있다. 이로인해 기존의 딥러닝으로는 새로운 센서나 부품이 추가 될 경우 이를 판독하기 위해 다량의 데이터 셋이 추가적으로 필요하다.

이러한 문제를 해결하기 위해 여러 기법을 동원하여 한장 혹은 몇 장의 극 소량의 이미지 데이터 만으로도 새로 추가되는 부품을 판독해내는 기존의 딥러닝 방식의 개선해보고자 이번 졸업 과제로 선정하게 되었다.



[그림1 임베디드 기반의 딥러닝 사용 분야]

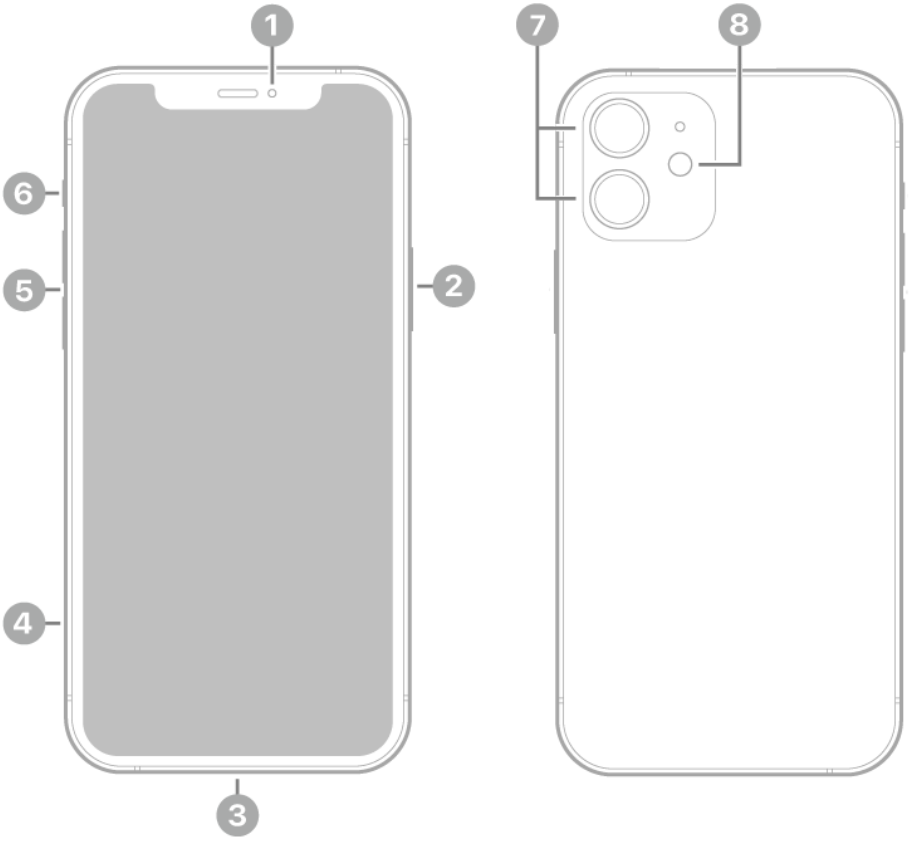
1.2 과제 목표

본 졸업 과제는 수집할 수 있는 데이터의 양이 적은 환경에서 다양한 기법, 예를 들면 전이학습이나 데이터 증강 등을 이용하여 충분한 정확도를 보이는 모델을 제작하는 것을 목표로 한다. 또한 Tiny ML의 특성상, 모델을 제한된 자원 환경에서 구동될 수 있을 만큼 저전력, 경량화 하는 것이 전제 될 것이다. 이와 동시에, 추가되는 파트의 경우, 한장 혹은 아주 극소량의 데이터셋으로 이를 정확하게 판별해내는 것을 보여야 한다.

2. 요구 조건 분석

2.1 기능적 요구 사항

1차적으로 부품의 종류를 구분해야 한다. 일상 생활에서 주로 고장나거나 파손으로 교환이 자주 이루어지는 부품을 선별하여 모델을 만들 예정이다. 이러한 부품들을 판별하는 모델은 정확도 높은 판독이 진행되어야 할 것이다. 이렇게 학습시킨 모델은 Tiny ML 성격에 맞게 충분히 경량화와 최적화를 진행해야 한다.



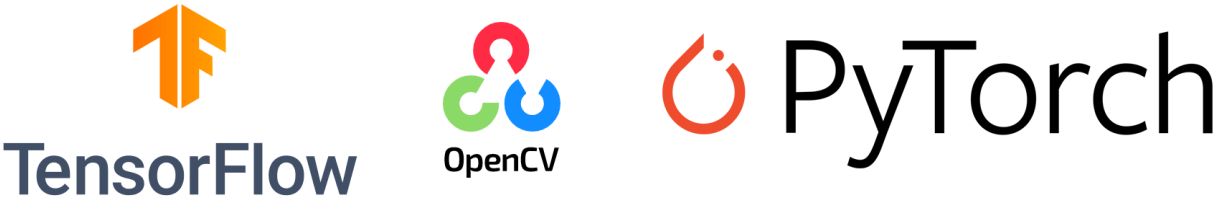
[그림2 예상 되는 파손 및 고장나는 부위 예]

2.2 진행 방안

2.2.1 모델 학습의 진행

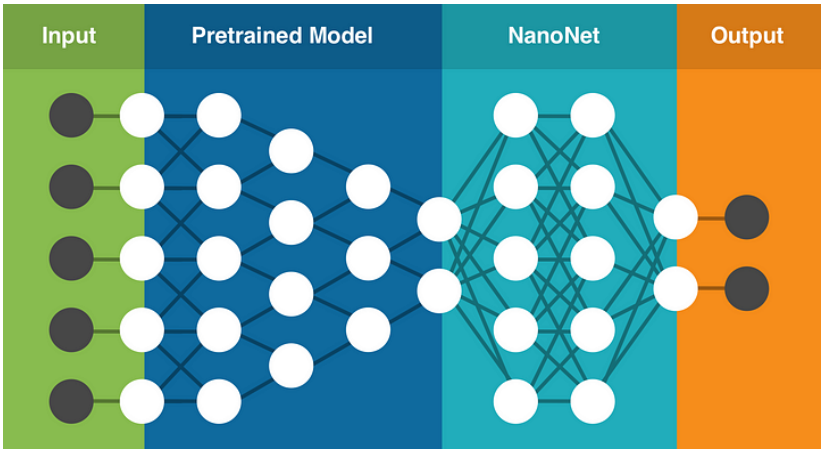
이번 프로젝트에서 가장 중요한 것은 부품을 판별하는 모델을 제작하는 것이다. 판별의 정확도를 테스트 하기 위해 먼저 4종류 정도를 판별할 수 있는 임시 테스트 모델을 제작할 예정이다. 이 때, 모델을 학습시킬 자료로, 우선 크기가 어느정도 있는 인식에 용의한 부품을 이용하여 제작을 시도해볼 예정이다.

다음으로 촬영된 이미지의 처리를 진행할 프레임 워크를 선택해야 한다. TensorFlow, OpenCV, PyTorch 등 다양한 프레임 워크가 있다. 이중 TensorFlow를 이용하여 테스트 모델의 제작을 시도할 예정이다. 이를 통해 인식의 정확도가 어느정도인지, 환경은 어떻게 구성해야될지 계획한다. 이 테스트를 바탕으로 정확도가 높은 모델을 만드는 것이 첫번째 과정이다.



[그림3 이미지 처리를 위한 오픈소스 프레임워크들]

정확도를 높이기 위해 많은 데이터를 학습시키는 것이 가장 좋은 방법이다. 하지만 이번 목표는 적은 데이터로 충분한 데이터를 학습시킨 것과 유사한 효과를 얻는 방법을 찾으려고 한다. 해결책으로 Transfer Learning (전이 학습) 을 이용하여, 좀 더 적은 양의 데이터로 정확도를 높일 수 있을 것으로 보인다. 전이 학습이란 하나의 작업에 대해 훈련 된 모델로 데이터 유형의 관계를 포착하고 동일한 도메인의 여러 문제에 대해 쉽게 재사용하는 방식이다. 그 중 한 예시로 NanoNets 모델이 있다. 이 모델은 GPU를 요구하지 않기에, 제한된 임베디드 하드웨어 환경에서 사용하기 적합해 보인다.



[그림4 NanoNets의 전이 학습 과정도]

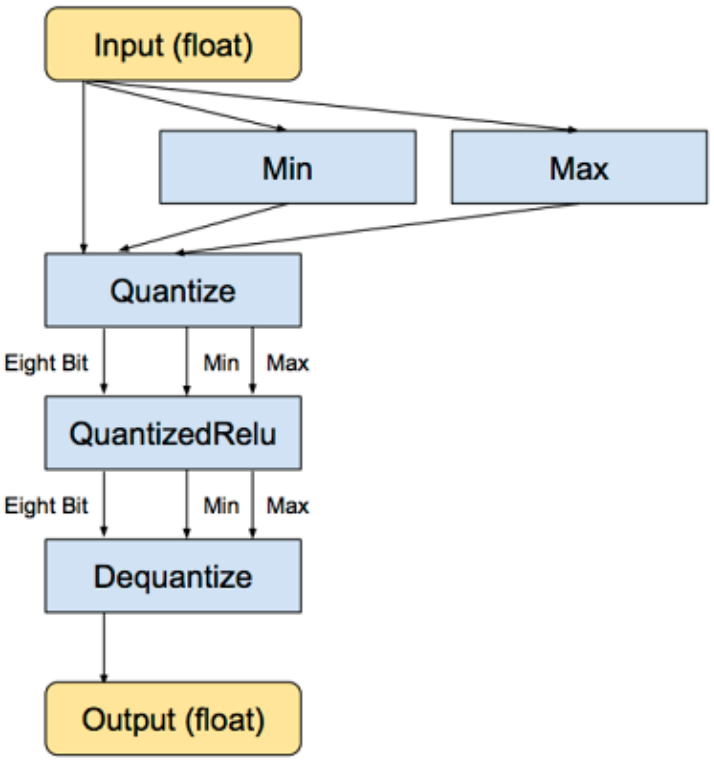
2.2.2 완성된 모델의 경량화

완성한 모델은 임베디드 환경보다 훨씬 고성능에서 사용할 수 있도록 만든 모델이기에, 바로 임베디드 컴퓨터에서 구동하기엔 무리가 있다. 따라서 다음 목표는 딥러닝 모델을 임베디드 환경에서 사용할 수 있을 만큼 최적화를 진행해야 한다.

최적화 방법으로는 학습된 모델을 경량화를 진행하거나 외부에서 연산후 결과만 받아오는 방법이 있다. 이번 프로젝트에서는 임베디드만을 이용하여 판별을 진행하고자 하기에 모델의 경량화를 진행해야 한다.

경량화는 크게 두 관점에서 볼 수 있다. 하드웨어적인 관점에서는 Network Quantisation, Network Compiling 등과 같은 방법이 있다. 다른 관점으로는 네트워크 구조적인 관점의, Efficient Architecture Design, Network Pruning, Knowledge Distillation, Matrix / Tensor Decomposition 등과 같은 방법이 있다. 이번 프로젝트에서는 각 관점을 모두 적용하여 모델의 최적화를 진행해보고자 한다.

먼저 하드웨어적인 관점의 방법으로 양자화를 통해 weight값의 사용 비트를 줄여서 모델 크기를 줄인다. 보통은 float32 데이터 타입으로 네트워크 연산 과정이 표현 되는데, 이를 그 보다 더 작은 크기의 데이터 타입으로 변환하여 연산 후, 다시 float32 형태로 변환한다.



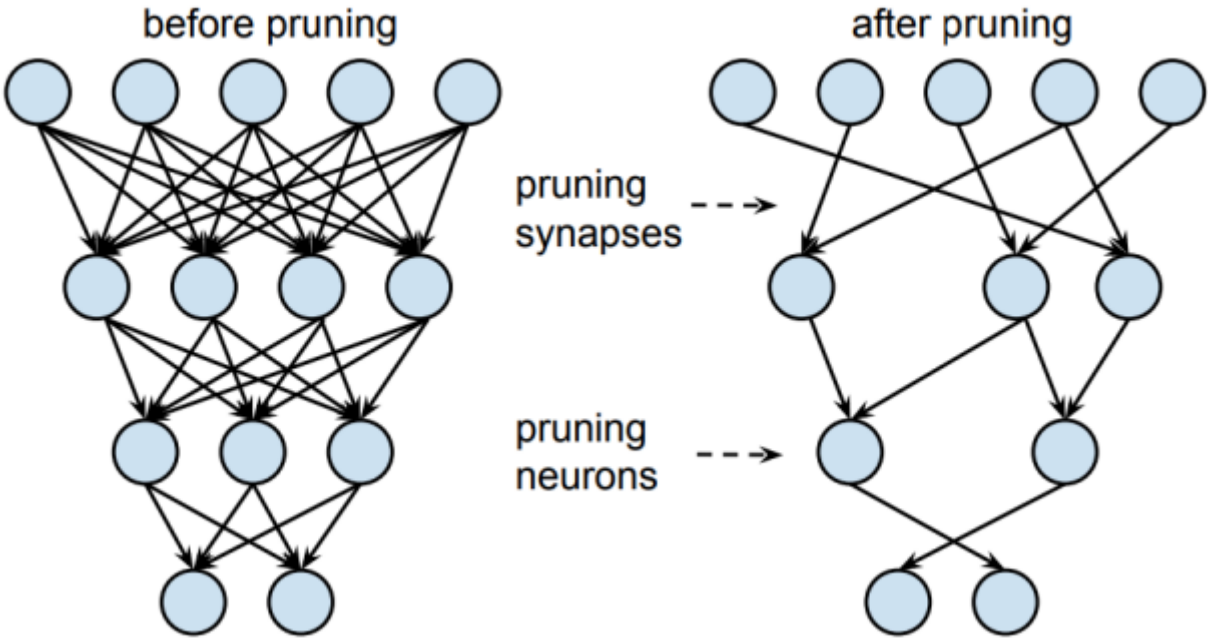
[그림5 양자화를 통한 크기 축소 모델]

이러한 과정을 거치면 최적화된 모델의 크기는 약 1/4로 감소하고, 정확도는 약간 하락하지만 정확도에 큰 영향을 미치지 않을 것으로 판단된다. 이미지 처리에 사용할 TensorFlow 프레임 워크의 경우 양자화를 통해 크기를 줄일 수 있음을 확인했기에, 이를 이용하여 진행하면 최적화에 도움이 될 것으로 보인다.

Model	Top-1 Accuracy (Original)	Top-1 Accuracy (Post Training Quantized)	Top-1 Accuracy (Quantization Aware Training)	Latency (Original) (ms)	Latency (Post Training Quantized) (ms)	Latency (Quantization Aware Training) (ms)	Size (Original) (MB)	Size (Optimized) (MB)
Mobilenet-v1-1-224	0.709	0.657	0.70	124	112	64	16.9	4.3
Mobilenet-v2-1-224	0.719	0.637	0.709	89	98	54	14	3.6
Inception_v3	0.78	0.772	0.775	1130	845	543	95.7	23.9
Resnet_v2_101	0.770	0.768	N/A	3973	2868	N/A	178.3	44.9

[그림6 양자화가 진행된 모델의 결과]

또 다른 모델 경량화의 방식으로 네트워크 구조적인 관점인 가지치기(Pruning) 방식을 이용할 수 있을 것이다. 가지치기란 Pruning은 네트워크의 수많은 weight들 중 상대적으로 중요도가 낮은 weight의 연결을 삭제하여 사용하는 파라미터의 수를 줄이고 네트워크를 경량화 하는 방법이다. 정확도의 손실 거의 없이 크기만 줄일 수 있는 장점이 있다. 이러한 방식을 이용하여 모델의 크기를 줄인다.



[그림7 프루닝이 진행되는 과정]

TensorFlow의 경우 Keras 라이브러리를 이용하면 이러한 가지치기 과정을 진행할 수 있다. 참조 문서에 따르면 프루닝 진행후의 정확도는 약간의 하락은 있지만 큰 차이가 나지 않는 것을 확인할 수 있다. 이러한 방식을 통해, 기존의 정확도가 높고 크기가 큰 고성능 환경에서 제작된 판별 모델을 임베디드 환경(라즈베리 파이)에서 사용할 수 있을 만큼 충분히 최적화 된 모델로 만들어 낸다.

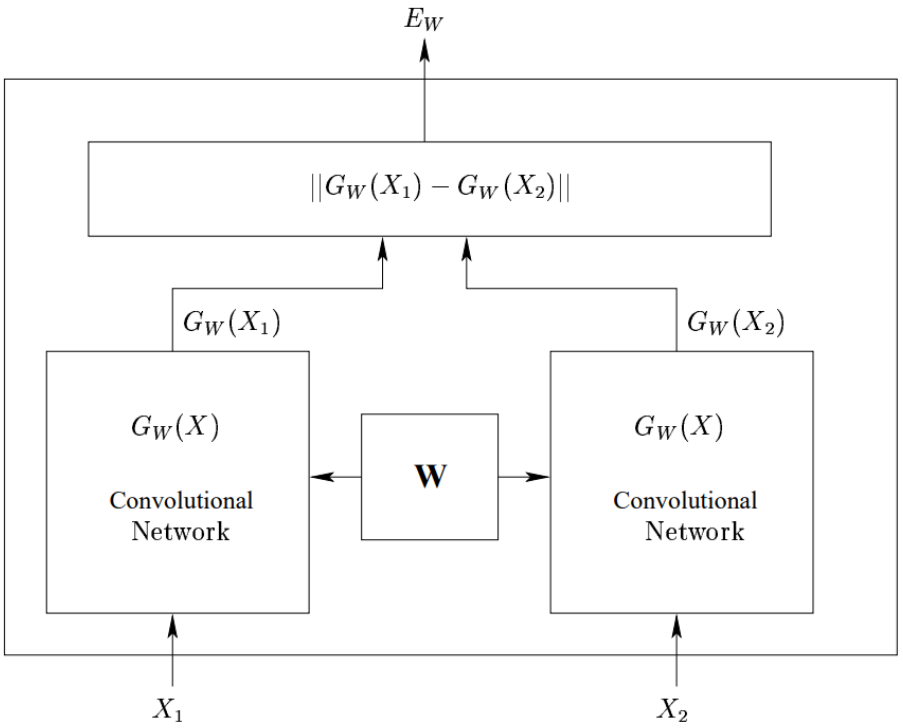
이미지 분류

모델	비희소 최상위 정확도	희소 정확도	희소성
InceptionV3	78.1%	78.0%	50%
		76.1%	75%
		74.6%	87.5%
MobilenetV1 224	71.04%	70.84%	50%

[그림8 Imagenet에서의 테스트 결과]

2.2.3 기종 추가

극소량의 이미지 데이터로 추가된 모델을 판별하기 위해, 적용할 수 있는 방법으로 One-Shot-Learning 혹은 Few-Shot-learning이라는 기법을 이용할 수 있을 것으로 보인다. 이때 Siamese Network의 설명을 참조하면 도움이 될 것으로 보인다. 샴 네트워크는 다루어야하는 클래스의 종류가 매우 많고, 특정 클래스에 대한 사진을 대량으로 구할 수 없을 때 머신러닝을 활용하여 그 클래스를 구분해내기 위하여 고안된 네트워크이다.

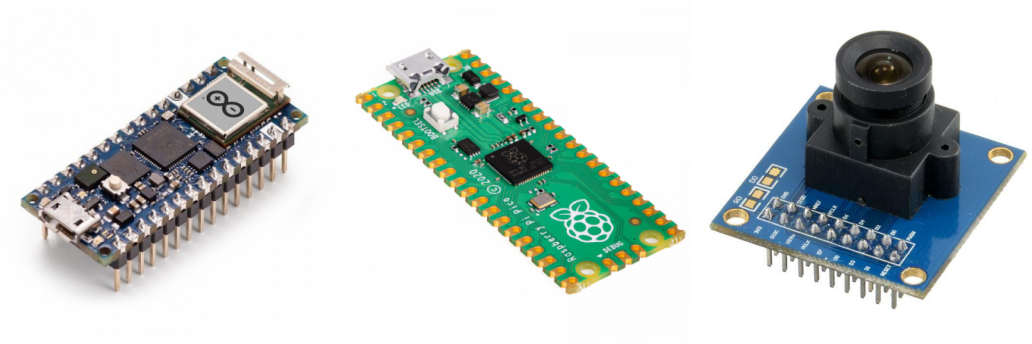


[그림9 샴 네트워크 구조]

기존 데이터와 유사하지만 다른 데이터를 활용하여 두 데이터 간의 유사도 가중치를 비교하여 동일 여부를

결정한다. 이 방식을 통해 매우 제한된 양의 데이터만을 사용하여 두 대상이 동일한지를 식별할 수 있다. 이러한 접근법은 데이터가 매우 적거나 희귀한 경우에 유용하며, 새로운 사물을 이전에 본 적 없는 상황에서도 유용하게 적용될 수 있다.

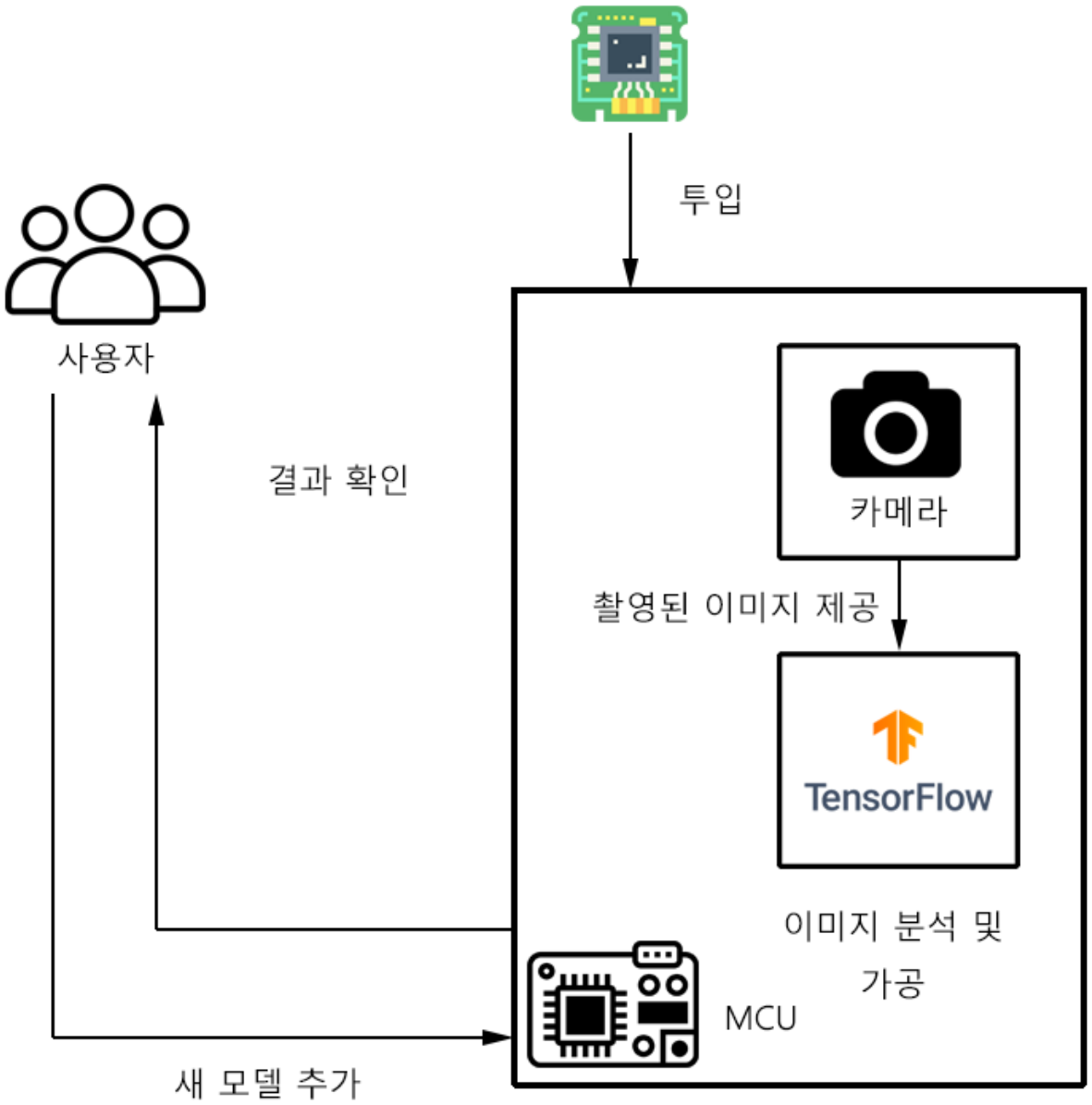
2.2.4 장치 구성



[그림10 아두이노 나노, 라즈베리 파이 피코, OV7670]

- 카메라 모듈: VGA OV7670 Camera Module은 저전력으로 작동 가능한 카메라이다.
- MCU: 후보 모델로 라즈베리파이 피코와 아두이노 나노가 있다. 사전에 모델을 만들어보고 이 결과를 바탕으로 기자재 선정을 진행할 예정이다.
- 기타 : 버튼 및 기타 자재는 추후 설계 후 크기에 맞춰 알아볼 예정이다,

2.2.5 사용 흐름도



[그림11 흐름도]

3. 예상 문제점

3.1 제약 사항

3.1.1 기존 데이터의 부재

판별을 진행할 부품의 이미지 데이터 수집은 어렵지 않을 것으로 보인다. 다만 기존에 존재하지 않는 데이터이므로 직접 이러한 이미지 데이터를 수집하고, 라벨링을 하는 등의 정제 과정을 거쳐 데이터 셋을 구축할 필요가 있어 보인다.

3.1.2 적용할 데이터의 선별

Tiny ML을 진행하고자 하기에 MCU의 자원에 제약이 매우 클 것이다. 이에 맞춰서 판별을 진행할 기종과 내부 부품의 종류를 결정해야 할 것으로 보인다. 초기에 큰 틀에서 어느정도의 모델 판별을 진행할지 어림잡고, 이후에 모델 경량화를 진행하여 여유 자원의 확보에 따라 유동적으로 조절해야 할 것으로 보인다.

3.1.3 판별의 정확도 문제

정제되지 않은 데이터로 학습할 경우, 판별의 정확도가 떨어지는 문제가 있다. 이를 해결하기위해, 촬영 환경의 통제와 통제된 환경에서의 정제된 데이터 수집을 진행할 예정이다. 해상도가 낮은 이미지를 바탕으로 제작할 예정이므로, 광원이 충분히 보장되어야 정확도 면에서 유리할 것으로 보인다.

4. 개발 일정 및 역할 분담

4.1 개발 일정

<table>
<tr><td colspan="2">5월</td><td colspan="5">6월</td><td colspan="5">7월</td><td colspan="5">8월</td><td colspan="3">9월</td></tr>
<tr><td>4주</td><td>5주</td><td>1주</td><td>2주</td><td>3주</td><td>4주</td><td>5주</td><td>1주</td><td>2주</td><td>3주</td><td>4주</td><td>5주</td><td>1주</td><td>2주</td><td>3주</td><td>4주</td><td>5주</td><td>1주</td><td>2주</td><td>3주</td></tr>
<tr><td colspan="3">사전 테스트
모델 개발</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="3">기자재
선정</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td colspan="4">Tiny ML 관련기술 학습</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td colspan="4">모델 경량화, 원 숏 러닝 및
데이터 증강 학습</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="7">실 사용 모델 개발</td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>중간
보고서</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="6">모델 테스트 및 보완 수정</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">외장 및 코드 제작</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="2">최종
보고서</td></tr>
</table>

4.2 역할 분담

이름	역할
윤태훈	임시 모델 개발 및 테스트 실 판별 진행용 부품 선정 및 수집 이미지 임시 전처리 코드 개발
이강윤	사용자에게 제시할 결과 데이터 전달 시스템 구축 실 사용을 위한 모델 경량화 및 개선 실기기에서의 이미지 수집 및 테스트
공통	데이터 전처리 부품 판별 모델 개발 보고서 작성 기기 테스트 기기 외장 제작 및 코드 작성

참조

http://daddynkidsmakers.blogspot.com/2019/01/blog-post.html
https://github.com/NanoNets/RaspberryPi-ObjectDetection-TensorFlow
https://medium.com/nanonets/nanonets-how-to-use-deep-learning-when-you-have-limited-data-f68c0b512cab
https://velog.io/@woojinn8/LightWeight-Deep-Learning-0.-%EB%94%A5%EB%9F%AC%EB%8B%9D-%EB%AA%A8%EB%8D%B8-%EA%B2%BD%EB%9F%89%ED%99%94
https://velog.io/@woojinn8/LightWeight-Deep-Learning-1.-Pruning
https://www.tensorflow.org/model_optimization/guide/pruning?hl=ko
https://www.hellodd.com/news/articleView.html?idxno=68801
https://amber-chaeeunk.tistory.com/110
https://3months.tistory.com/507