

문서명	중간보고서	작성일	2018.05.04
작성자	Hidden Layer	지도교수	이충목

[캡스톤 디자인 프로젝트]

---

---

**(주) 포스-벽진  
자동화 설비 개선**

---

---

팀 Hidden Layer	
학번	이름
201302455	이승진(PM)
201300061	강인선
201302032	오익준
201502574	이정하

# 목 차

1. 프로젝트 요약문 .....	3
2. 프로젝트 소개 .....	4
2.1 프로젝트 주제 및 필요성 .....	4
3. 프로젝트 내용 .....	6
3.1 Error Prevention System .....	6
3.2 Error Catcher System .....	8
4. 프로젝트 추진 결과물 .....	25
5. 프로젝트 기대 효과 .....	34
5.1 개발환경 .....	34
5.2 프로젝트 기대 효과 .....	34
5.3 향후 과제 및 최종 결과물 활용 계획 .....	34
6. 프로젝트 수행 후기 .....	35
7. 참고문헌 .....	37

## 1. 프로젝트 요약문

프로젝트 주제	(주) 포스-벽진 자동화 설비 개선
프로젝트 배경	<ul style="list-style-type: none"><li>- 자동화 설비를 도입한 (주) 포스-벽진</li><li>- 무인환경에서 금형 파손 시 대처 시스템 부재</li><li>- 실시간으로 금형 파손을 인식할 수 있는 시스템이 요구됨</li><li>- 금형 파손을 예측할 수 있는 시스템이 요구됨</li></ul>
프로젝트 목표	<ul style="list-style-type: none"><li>- 무인환경에서 금형 파손 시 작업자에게 알림을 보내는 시스템 구현</li><li>- 금형 파손 데이터를 기록하여 차후 예측할 수 있도록 금형 추천 시스템 및 PUSH 알림 App 구축</li></ul>
프로젝트 내용	<ul style="list-style-type: none"><li>- 금형 추천 시스템<ul style="list-style-type: none"><li>• 설계도 data 를 통해 작업 시 각 금형의 punching 횟수들을 저장하고 알맞은 금형을 추천해주는 시스템 구축 &amp; 파손 예측 시스템 구축</li></ul></li><li>- 데이터 수집<ul style="list-style-type: none"><li>• 야간 작업시간 또는 무인시간동안 마이크를 통해 Punching Sound 녹음</li><li>• Python 을 이용한 데이터 전처리</li></ul></li><li>- Deep Learning<ul style="list-style-type: none"><li>• 금형 녹음 데이터를 통해 Training</li><li>• 학습된 모델을 통해 test</li><li>• 완성된 모델로 Punching Sound 정상 / 비정상 구별</li></ul></li><li>- 적외선 센서<ul style="list-style-type: none"><li>• Arduino 와 호환된 적외선 빔 센서를 통해 scrap 인식</li><li>• 블루투스 모듈을 통해 인식 결과 값 App 에 전송</li></ul></li><li>- 시스템 구현<ul style="list-style-type: none"><li>• 금형 추천 시스템(App) &amp; Error Alarm App 구축</li></ul></li></ul>
프로젝트 최종 결과물의 활용계획	<ul style="list-style-type: none"><li>- 2학기 이공계 학술제 출품</li><li>- 교외 대한산업공학회 공모전 출마</li><li>- (주)포스-벽진 사에 시스템 개선 및 App 구축</li></ul>

## 2. 프로젝트 소개

### 2.1 프로젝트 주제 및 필요성

- 프로젝트 주제 : 자동화 설비 개선

#### 1) Error Prevention (EP) System 구축

주간&야간 작업 시 추천 로직을 통해 선택된 금형을 사용하여 Error 를 대비하기 위한 금형 추천 시스템. 총 26 가지의 금형을 각각 3 개씩 유지하여 생산할 때 금형을 추천한다.

#### 2) Error Catcher (EC) System 구축

야간 작업 시 발생한 Error 를 작업자에게 바로 알리고 대처하기 위한 시스템. 철판 scrap 이 떨어지는 여부를 판단하는 적외선 인식기와 punching 소리로 비정상과 정상 소리를 판단하는 음성인식을 종합하여 에러를 판단하고, 에러 발생 시 App 을 통해 알람을 보낸다.



야간 작업시 Min. Punching Time 의  
금형을 사용하여 Error를 대비하기 위한

**Error Prevention (EP System)**



야간 작업시 발생한 Error를  
작업자에게 바로 알리고 대처하기 위한

**Error Catcher (EC System)**

- 프로젝트 필요성

현재 야간 생산량 = 15days / month

평균 불량 횟수 = 2.5 days / month

금형의 종류 = 총 26 개

금형의 비용 = 2,284,000 (₩) (26 개의 금형의 총합)

상태에서 예방 관련하여 모든 금형을 교체하는 것을 고려했을 때,

나오는 경우의 수와 각 대안의 비교는 다음과 같다.

경우의 수	비용	비교 분석
A. 1 일에 한번씩 금형 교체	고정비(3426 만원) + 금형재고비용 (악성 재고)	에러가 나올 확률이 없지만, 막대한 재고 비용과 복잡한 재고관리가 요구된다.
B. 3 일에 한번씩 금형 교체	고정비(1142 만원) + 금형재고비용(악성 재고)	A의 대안보다는 비용이 적지만 여전히 높은 재고 비용과 복잡한 재고관리가 요구된다.
C. 5 일에 한번씩 금형 교체	고정비(685 만원) + 금형재고비용(악성 재고)	회사에서 해본 결과 재고관리와 재고비용적인 측면에서 부담을 느껴서 중단하였다.
D. 15 일에 한번씩 금형 교체	고정비(228 만원) + 금형재고비용(악성 재고)	한달 오류 횟수를 충족하지 않은 경우이므로 고려하지 않는다.
E. 금형 추천 , error 발생 후 알림	1,171,500 원의 비용이 발생	금형 추천을 통해 error 발생 확률을 최대한 낮추고 알림을 통해 에러를 대처할 수 있게 때문에 가장 합리적인 대안이다.

<Table 1. 대안 비교 표>

C 대안을 포스 회사가 직접 적용 해 보았지만, 재고관리와 재고비용적인 측면에서  
부담을 느꼈다. 따라서 금형 추천을 통해 에러를 대비하고, 더 나아가 차후  
Error 상황을 대처하기 위한 알림 시스템이 필요하다.

### 3. 프로젝트 내용

#### 3.1 Error Prevention System (금형 추천 시스템)

금형에 작용하는 요인은 punching 횟수 & 철판 두께 이다.

Punching 횟수는 설계도 data 를 통해 각 금형의 총 punching 횟수를 계산할 수 있다.

Punch force 에 대한 식은 다음과 같다.

$$F = S \times t \times L$$

( $S$  = Shear strength(전단강도) ,  $t$  = Sheet thickness ,  $L$  = Length of cutting edge)

If shear strength is not known, cutting force can be estimated as :

$$F_{max} = (UTS) \times t \times L \div 1000$$

( $UTS$  = ultimate tensile strength of sheet metal(최대인장강도) ,  $t$  = thickness ,

$L$  = Length of cutting edge)

재료가 같은 steel 은 UTS 값이 같으므로 punching force 는 두께에 따라 결정된다.

위의 식을 이용하여 계산해 본 결과, 1.6T 와 2T 를 punching 하는 힘은 다음과 같다.

$$F_{1.6T} = UTS \times 1.6 \times L \div 1000$$

$$F_{2T} = UTS \times 2 \times L \div 1000$$

$$\rightarrow F_{2T} = (1.25) \times F_{1.6T}$$

2T 를 1 번 punching 하는 것은 1.6T 를 1.25 번 punching 하는 것과 같다.

따라서, 2T 를 생산할 때에는 punching 횟수에 1.25 를 곱하여서 계산한 값을 punching 횟수에 저장한다.

주간 생산일 경우에는 에러에 즉각 대비할 수 있으므로 2 번째로 적은 punching 횟수의 금형을 추천하고 야간에는 에러의 발생을 낮추기 위해 가장 적은 punching 횟수의 금형을 추천하도록 설계하였다.

매번 생산량을 입력할 때마다 26 개의 금형을 계속 바꾸는 것은 번거로우므로 효율성을 높이기 위해 주간과 야간에 한 번씩, 하루에 총 2 번 금형을 추천하도록 설계하였다.

EX)

펀처 종류	사용한 Punch 횟수
A_SQ_1	689
A_SQ_2	9849
A_SQ_3	29830

야간 → A\_SQ\_1 추천

주간 → A\_SQ\_2 추천

A\_RO 금형을 예시로 들면 다음과 같다.



처음 주간 생산에는 A\_RO\_1 번 금형을 추천한다.

다음 주간 생산에는 2 번째로 punching 횟수가 적은 A\_RO\_2 번 금형을 추천한다.

그 다음 주간 생산에는 punching 횟수가 2 번째로 적은 A\_RO\_1 번 금형을 추천한다.

야간 생산에는 punching 횟수가 가장 작은 A\_RO\_3 번 금형을 추천한다.

시간이 지나고 Punching 한도에 가까워져 가장 많이 punching 한 A\_RO\_1 번 금형이 파손되면, 금형을 교체하고 다시 A\_RO\_1 번 금형을 야간에 쓰이는 금형으로 바꿔게 된다.

### 3.2 Error Catcher System

#### ① 적외선 인식

떨어지는 철판 scrap 을 인식하기 위해 아두이노와 호환이 가능한 초소형 적외선 빔 센서를 사용하여 적외선 인식기를 만들었다.

빠른 속도로 떨어지는 scrap 의 인식률을 높이기 위해 인식기 사이에 종이를 설치하였다. 떨어지는 scrap 이 종이를 건드리게 되면 빔 센서의 수광, 발광 센서가 연결되어 인식이 되는 원리이다.



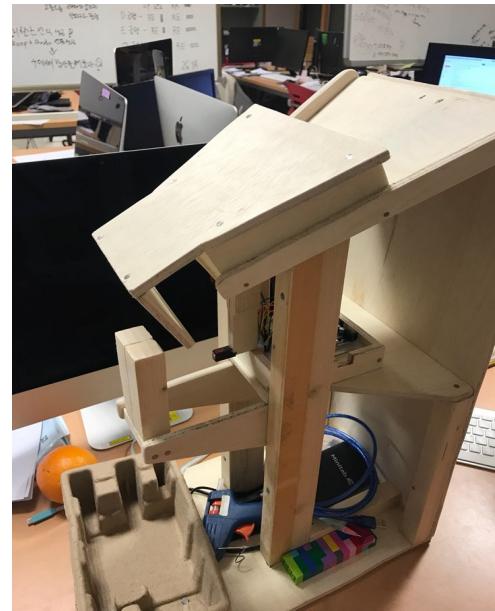
<Figure 1. 컨베이어 벨트>

위의 사진과 같이 컨베이어 벨트에서 scrap 이 떨어지는 범위는 25cm 로 적외선 센서가 이 모든 범위를 인식하기에는 한계가 있다.

따라서 인식의 정확도를 높이기 위해 인식하는 부분을 25cm 에서 10cm 로 좁히는 보조기구를 만들었다.



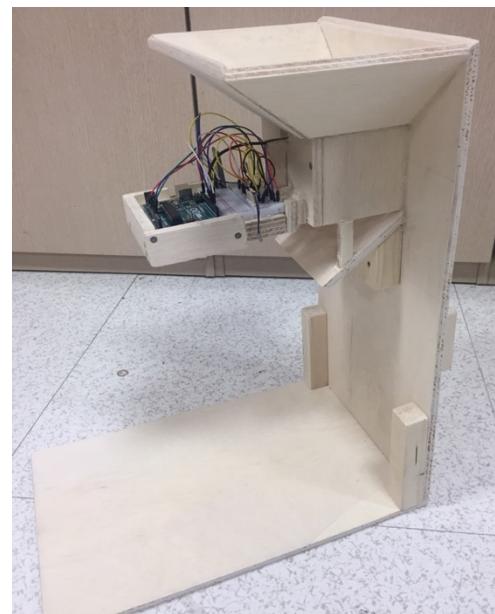
<Figure 2. 첫번째 보조기구>



<Figure 3. 두번째 보조기구>



<Figure 4. 세번째 보조기구>



<Figure 5. 네번째 보조기구(최종)>

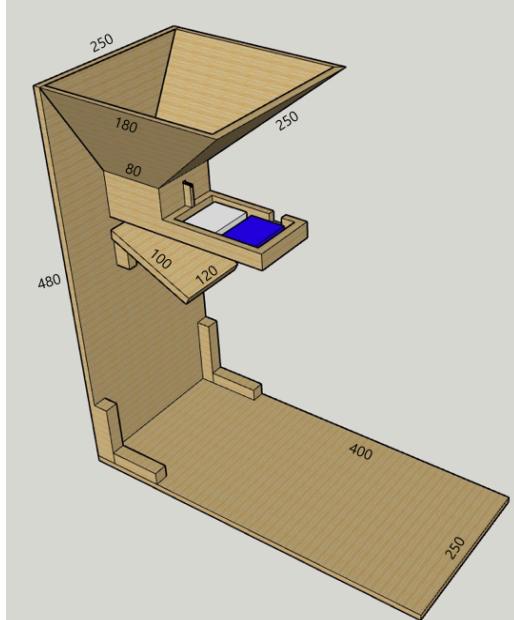
인식률을 최대한 높이기 위해 총 4 차례에 걸쳐서 보조기구를 만들었다.

첫번째 보조기구는 scrap 이 적외선 인식기를 거치지 않고 그냥 지나쳤다.

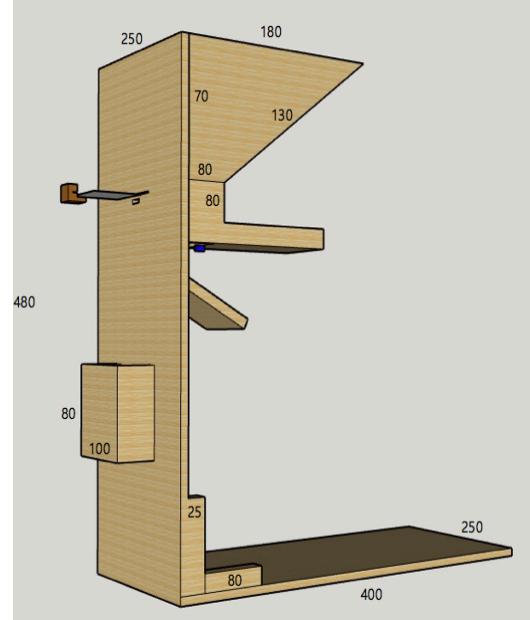
이 문제를 보완하기 위해 <Figure 3> 과 같이 적외선 인식기로 scrap 이 떨어지도록 유도하는 덮개를 만들었다. 이 보조기구를 설치하기 위하여 현장에 방문하였으나 보조기구의 높이가 잘못 측정되어 있어서 설치가 불가 하였다. 그래서 다시 측정한

높이에 맞추어 보조기구를 수정하였고, 재방문하여 설치를 시도했다. 하지만 scrap 이 적외선 인식기에 도달하지 못하는 결정적인 문제가 발생하였다.  
보조기구의 설계에 문제가 있음을 인식하고 재설계를 하였으며, 최종적으로 설치를 마무리 하였다.

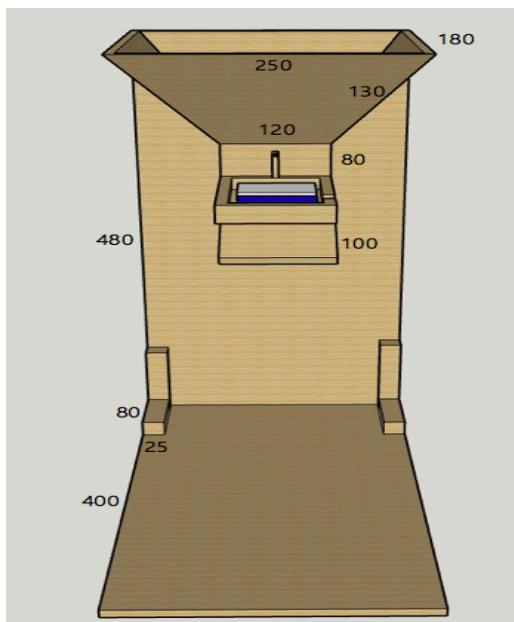
다음은 보조기구의 설계도이다.



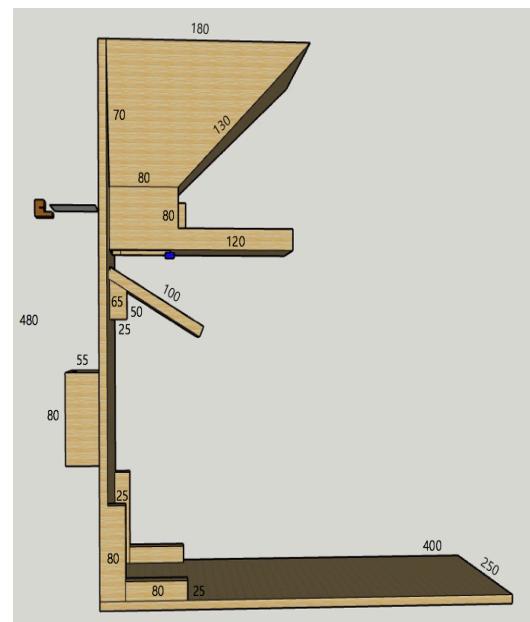
<Figure 6. 보조기구 설계도 1>



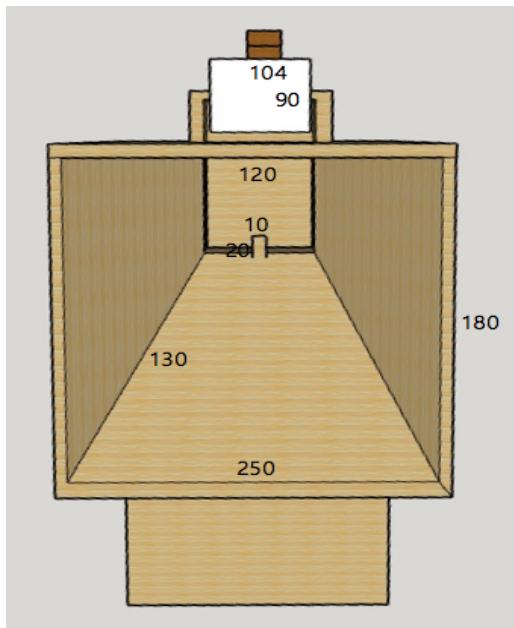
<Figure 7. 보조기구 설계도 2>



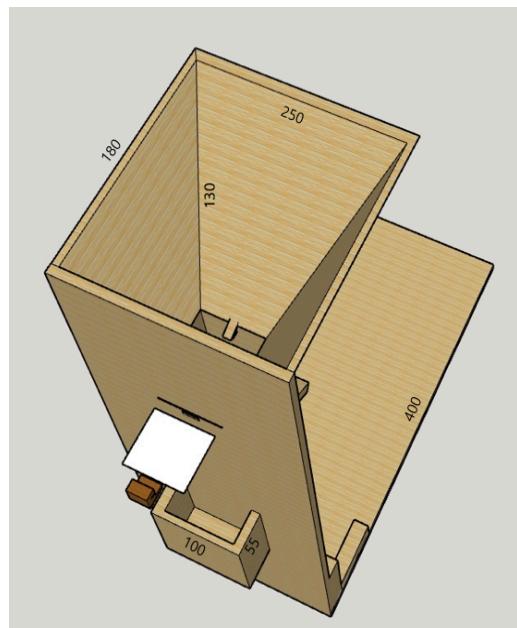
<Figure 8. 보조기구 설계도 3>



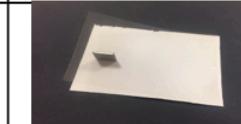
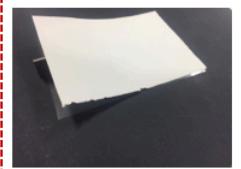
<Figure 9. 보조기구 설계도 4>



<Figure 10. 보조기구 설계도 5>



<Figure 11. 보조기구 설계도 6>

			
1. 종이 1장 30분 동안 작업 시, 형태 변형	2. 종이 1장 + 종이 ½장 받침대 2시간 동안 작업 시, 형태 변형	3. 코팅지 코팅지 위에 스크랩이 쌓이는 현상 발생	
			 종이 1장 + OHP 1장 받침대

<Table 2. 종이 실험>

적외선 인식기 사이의 종이의 종류를 선택하기 위해, 실험을 하였다.

- 1 번 : 종이 1장을 사용한 결과, 30 분 동안 scrap 이 떨어졌을 때 형태 변형이 일어났다.
- 2 번 : 종이 1장과 종이 반장을 받침대로 사용한 결과, 2 시간 동안 scrap 이 떨어졌을 때 형태 변형이 일어났다.
- 3 번 : 코팅지를 사용한 결과, 큰 scrap 이 떨어지는 경우에는 잘 인식하지만, 크기가 작은 scrap 들은 코팅지 위에 쌓이는 현상이 발생했다.
- 4 번 : 종이 1장과 OHP 필름 반장을 받침대로 사용한 결과, 2 시간 동안 scrap 이 떨어졌을 때 형태 변형이 일어났다.
- 5 번 : OHP 필름 1장과 종이 1장을 받침대로 사용한 결과, OHP 필름과 종이 사이 유격이 발생하였다.
- 6 번 : 종이 반장과 OHP 필름 1장을 받침대로 사용한 결과, 수평상태를 유지하지 못하는 현상이 발생했다.

종이 1장과 OHP 필름 1장이 받침대로 사용되었을 때 최적의 조건임을 발견하였고, 이 조건으로 실험 및 검증을 진행하였다.



<Figure 12. 1 일 동안 사용>



<Figure 13. 2 일 동안 사용>

위의 사진은 최적의 조건으로 선정된 종이 1 장과 OHP 필름 1장을 받침대로 사용하였을 때의 사진이다.

<Figure 12> 은 실제 현장에서 1 일 동안 사용한 종이이고 <Figure 13> 은 2 일 동안 사용한 종이이다. 3 일을 사용하게 될 경우 종이의 인식 부분이 찢겨서 인식의 정확도가 낮아지는 현상이 발생하였다. 따라서 적외선 인식기 보조기구 뒤에 종이 보관함을 만들어 작업자가 편리하게 종이를 바꿀 수 있도록 설계하였다.



<Figure 14. 종이 보관함>

## ② 음향 인식

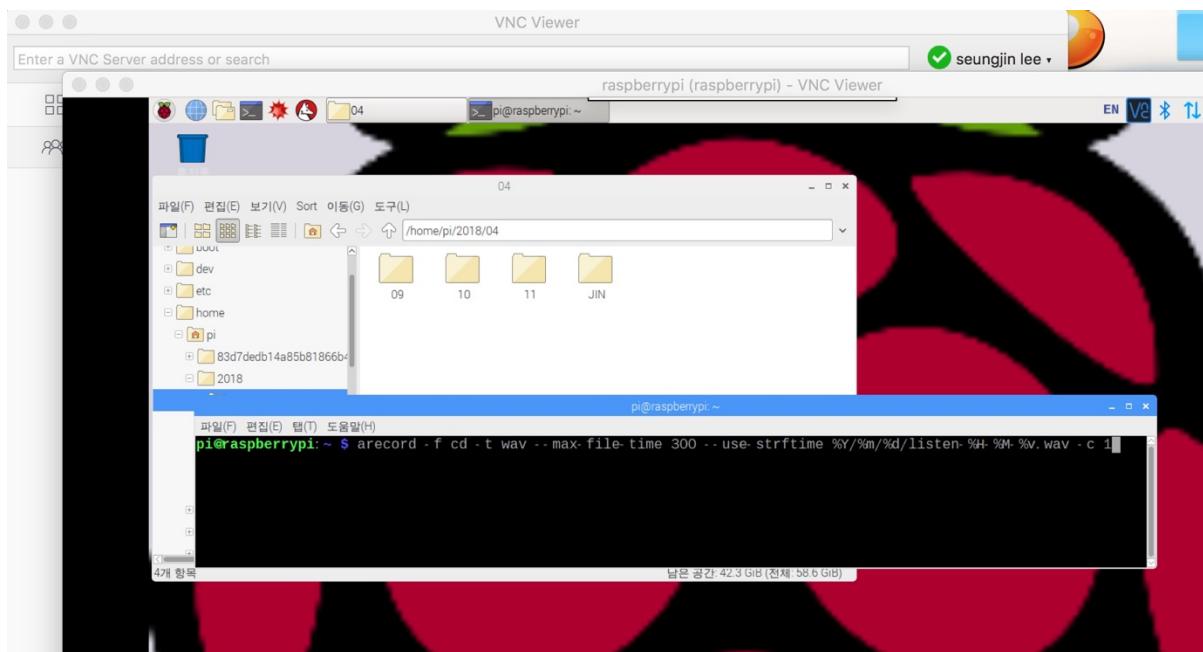
### - 데이터 수집

자동화 설비 옆에 마이크를 설치하고 설치된 마이크를 Raspberry Pi 와 연결하였다.

3 분 단위로 녹음을 하였으며, 원격 (VNC)을 사용해 공학관 502 호실에서 실시간으로 punching 소리 데이터를 수집하였다.



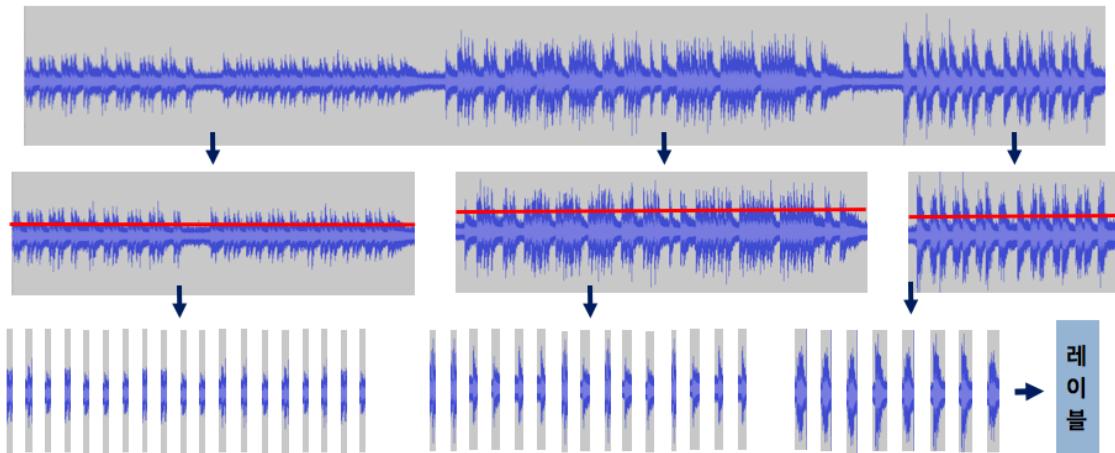
<Figure 15. 마이크를 설치한 현장 모습>



<Figure 16. VNC 원격>

### - 데이터 전처리

녹음된 punching 소리 데이터는 금형 별로 수동으로 편집되어 저장된다. 저장된 데이터들은 Python으로 구현한 소리 편집 프로그램으로 통하여 1 번의 punching 당 1 개의 소리 데이터로 자동적으로 편집되어 저장된다. 구현한 Python 소리 편집 프로그램은 특정 주파수 이상이면 편집되는 프로그램이다. 최종적으로 편집된 데이터는 금형 별로 labeling 되며 training 과정으로 넘어가게 된다.



<Figure 17. Data 전처리 과정>

### - Preprocessing

Unsupervised 기법 중 하나인 Generative Adversarial Network (GAN) 는 이미지 training 기법이기 때문에, wav 파일을 그대로 training 하는 것보다 이미지 파일로 바꿔서 training 하는 것이 더 적합하다. 따라서 Python 을 통해 wav 파일을 Mel-spectrogram 이미지 파일로 변환한다. 변환된 데이터들은 각 금형 별 이미지의 Mean , Standard deviation 값을 구하고 정규화 하여 training\_data.npz 파일에 저장된다.

### - Training

Batch 값, Hidden layer 수, 반복 횟수인 Epoch 값을 설정한 뒤 training 을 한다. Epoch 1 회마다 Generator 가 가짜 이미지를 만들며, 그것을 Discriminator 가 평가한다. Generator 가 그 전 보다 실제 이미지와 비슷한 이미지를 다시 만들고 Discriminator 가 평가한다. 이 과정을 설정한 Epoch 수 만큼 반복한다. TanH 과 ReLU 함수를 이용하여, Discriminator 가 정상이라고 생각하면 표현할 수 있는 만큼의 가장 큰 숫자를 Output 으로 나타내 주며, 비정상은 가장 작은 숫자로 나타내 준다.

### - Training 결과

502-1	502-2	502-3	502-4	GPU
Training 기간 : 7일 정확도 : 20% 아래 Epoch : 900 Hidden layer : 3 Batch : 64 → 폐기	Training 기간 : 8일 정확도 : 대략 50% Epoch : 970 Hidden layer : 3 Batch : 64 → 아직 진행중	Training 기간 : 7일 정확도 : 30% 아래 Epoch : 800 Hidden layer : 4 Batch : 64 → 폐기	Training 기간 : 8일 정확도 : 대략 85% Epoch : 1500 Hidden layer : 2 Batch : 64 → 진행중	Training 기간 : 1.5일 정확도 : 20% Epoch : 3000 Hidden layer : 4 Batch : 64 → 폐기  Training 기간 : 1.5일 정확도 : 대략 85% Epoch : 2500 Hidden layer : 2 Batch : 64 → 완료  Training 기간 : 진행중 정확도 : 아직 10% Epoch : 600 Hidden layer : 3 Batch : 32

<Table 3. Training 결과>

2 주 동안 502 호실 컴퓨터 4 대와 별도로 GPU 를 함께 training 하였다.

그 중 결과값이 가장 좋은 모델(502-2 model)을 선택하여 실험을 진행하였다.

5월 15일부터 5월 28일, 총 9일 동안 실험을 진행하였으며 정확도는 다음과 같다.

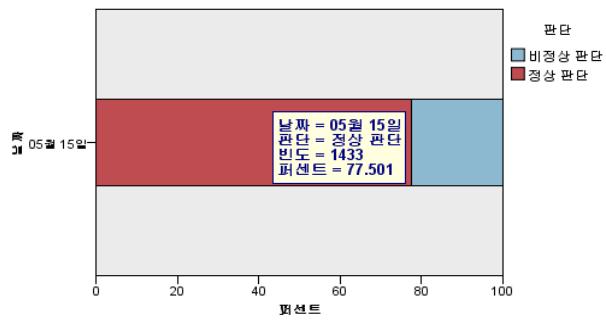


<Figure 18. Training 전 정확도>

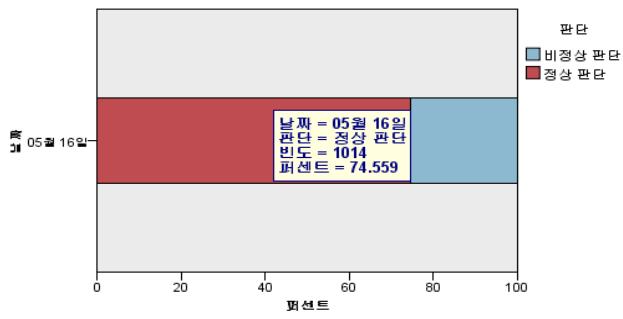
9일간의 정확도는 77.03%이며, 총 Data taking 수는 10665 개이다.

날짜	Data taking 수	정확도
5월 15일	1849	77.50%
5월 16일	1360	74.55%
5월 17일	1816	77.03%
5월 18일	985	78.11%
5월 21일	252	84.92%
5월 23일	1067	78.35%
5월 24일	1279	74.82%
5월 25일	1226	77.97%
5월 28일	836	76.62%

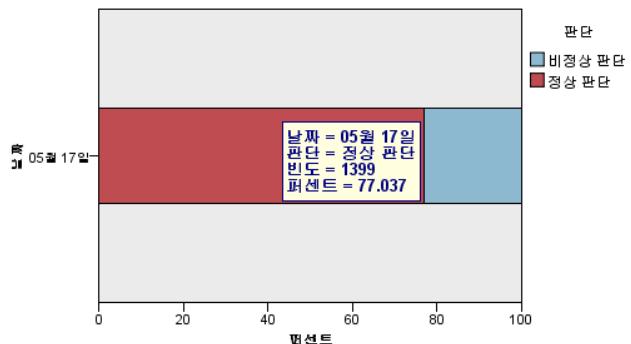
<Table 4. Training 전 정확도>



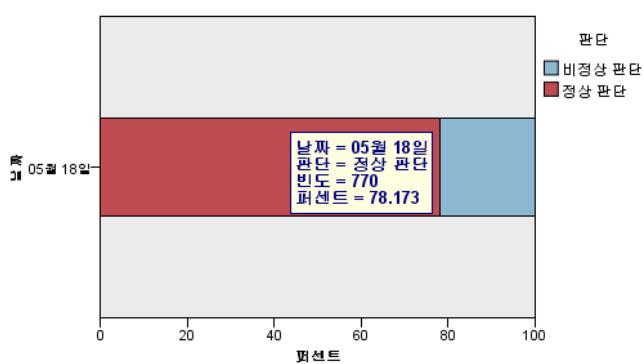
<Figure 19. 5/15 정확도>



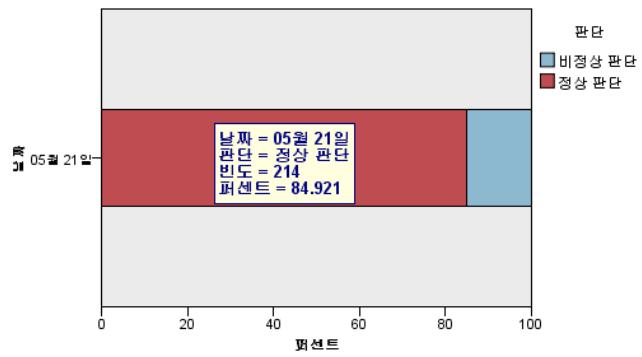
<Figure 20. 5/16 정확도>



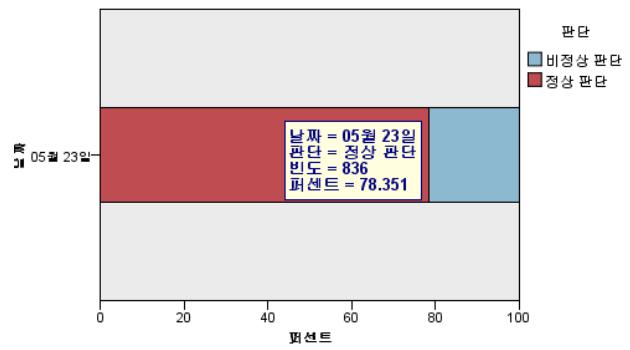
<Figure 21. 5/17 정확도>



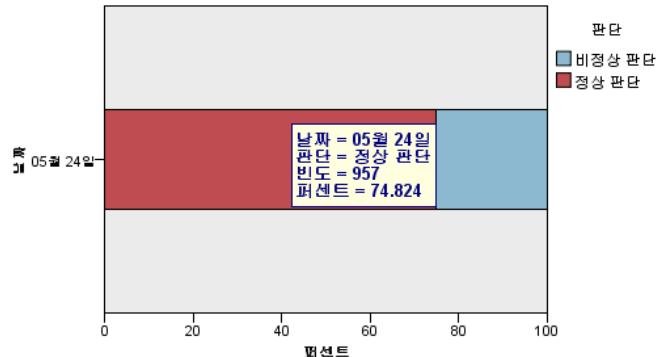
<Figure 22. 5/18 정확도>



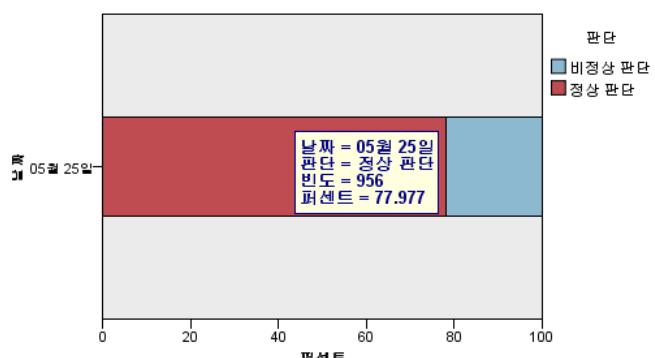
<Figure 23. 5/21 정확도>



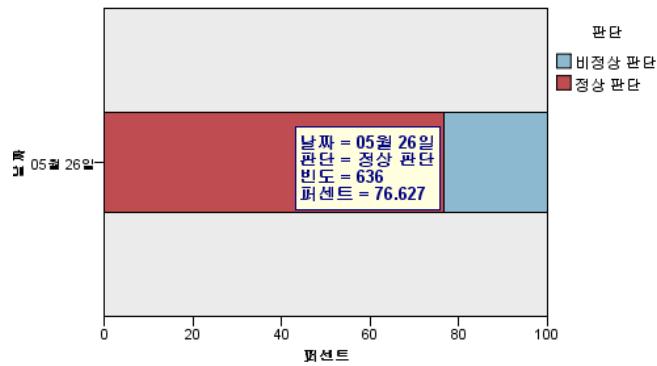
<Figure 24. 5/23 정확도>



<Figure 25. 5/24 정확도>



<Figure 26. 5/25 정확도>



<Figure 27. 5/28 정확도>

정확도가 77.03%로 낮은 결과가 도출되었다.

Training 하지 않았던 금형이 최근에 사용되어 그 금형을 비정상이라고 인상하는 현상이 있었고 따라서 그 금형을 포함하여 새로 training 을 했으며, 다시 training 한 model 로 실험한 결과는 다음과 같다.

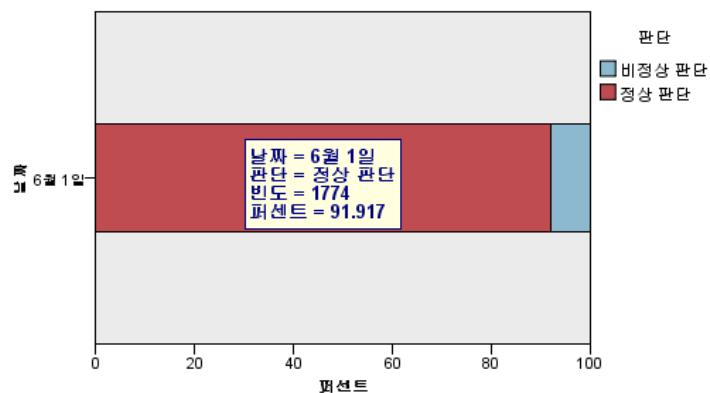
6 월 1 일부터 6 월 7 일, 총 4 일 동안 다시 training 한 model 로 실험을 진행하였으며 정확도는 다음과 같다.

필드	일치 비율	%	빈도
판단	92.21	5527	

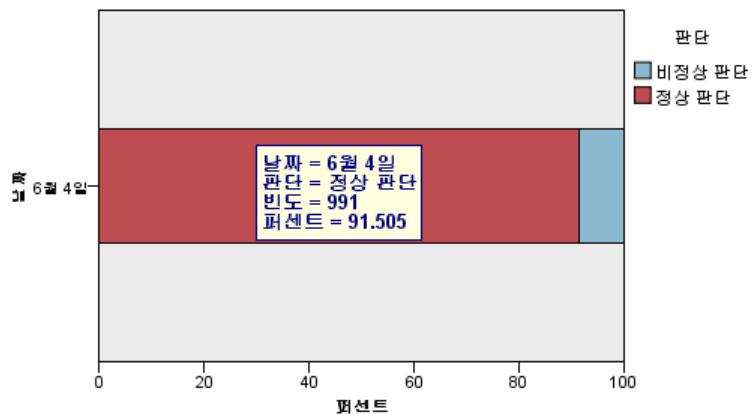
<Figure 28. Training 후 정확도>

날짜	Data taking 수	정확도
6월 1일	1931	91.91%
6월 4일	1083	91.50%
6월 5일	1304	93.32%
6월 7일	1677	92.12%

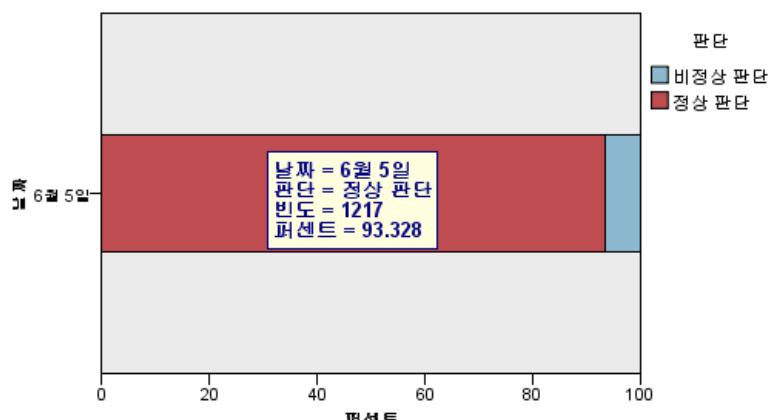
<Table 5. Training 후 정확도>



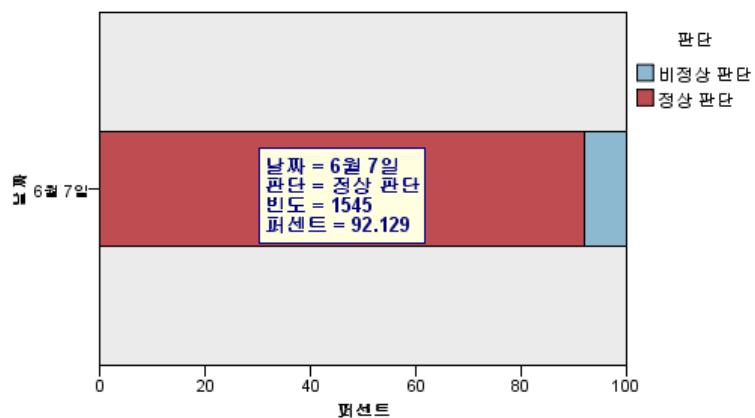
<Figure 29. 6/1 정확도>



<Figure 30. 6/4 정확도>



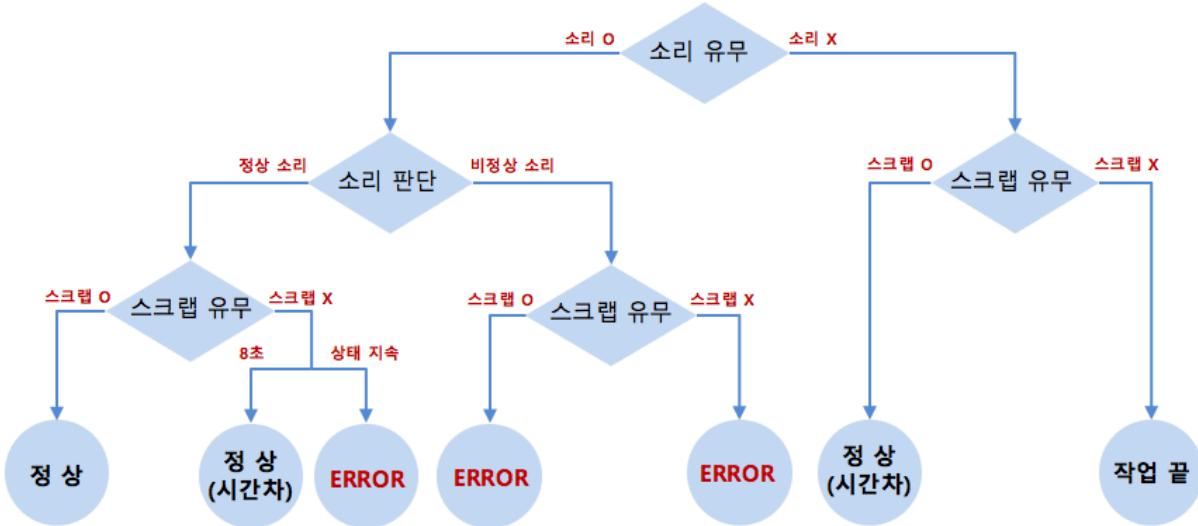
<Figure 31. 6/5 정확도>



<Figure 32. 6/7 정확도>

정확도가 92.21%로 목표정확도에 도달하였다.

### ③ Error Catcher 시스템 의사결정



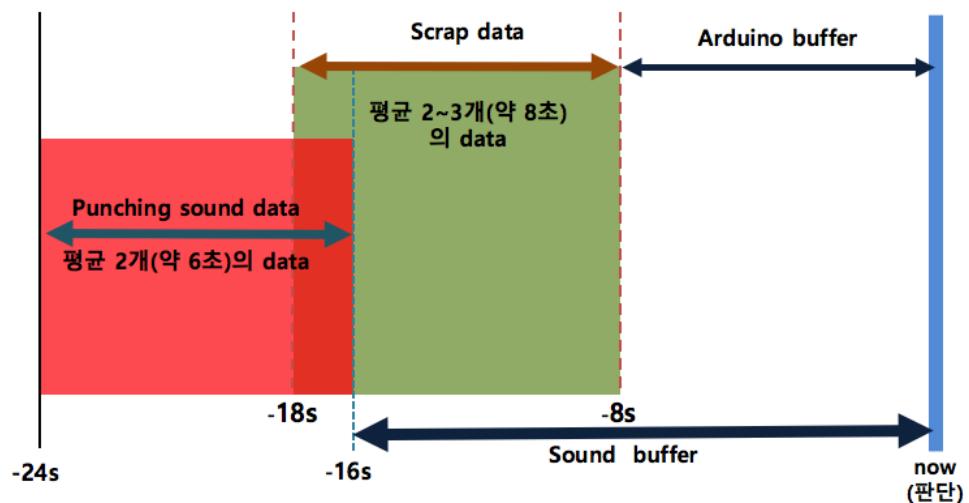
<Figure 33. 의사결정 로직>

Error Catcher 시스템의 의사결정 로직은 위와 같다.

Error 라고 판단되는 경우는 총 3 가지가 있다.

첫번째, 정상 소리가 나지만 스크랩이 떨어지지 않는 상태가 지속될 경우.

두번째, 비정상 소리가 나는 경우이다.



<Figure 34. 의사결정>

위의 그림은 의사결정 방법이다.

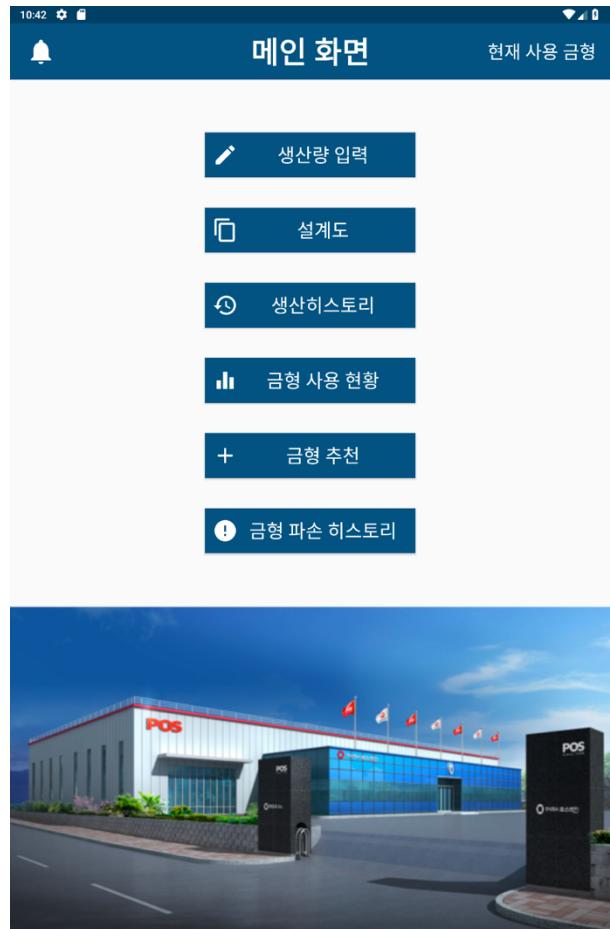
적외선 인식은 편침 소리가 난 뒤에 컨베이어 벨트를 따라 떨어지는 데 까지 약 8 초의 시간이 소요된다. 따라서 먼저 편침 소리 data 를 수집하고 8 초 뒤에 scrap data 를 수집한다. Buffer 구간을 두어 그 기간 동안 지속적으로 에러인지 아닌지 판단을 하며, 에러라고 판단해도 에러 상태가 4 번 축적될 때 알람을 보내도록 설계하였다.

## 4. 프로젝트 추진 결과물

### 4.1 Error Prevention System



Error Prevention System 메뉴구조도이다.



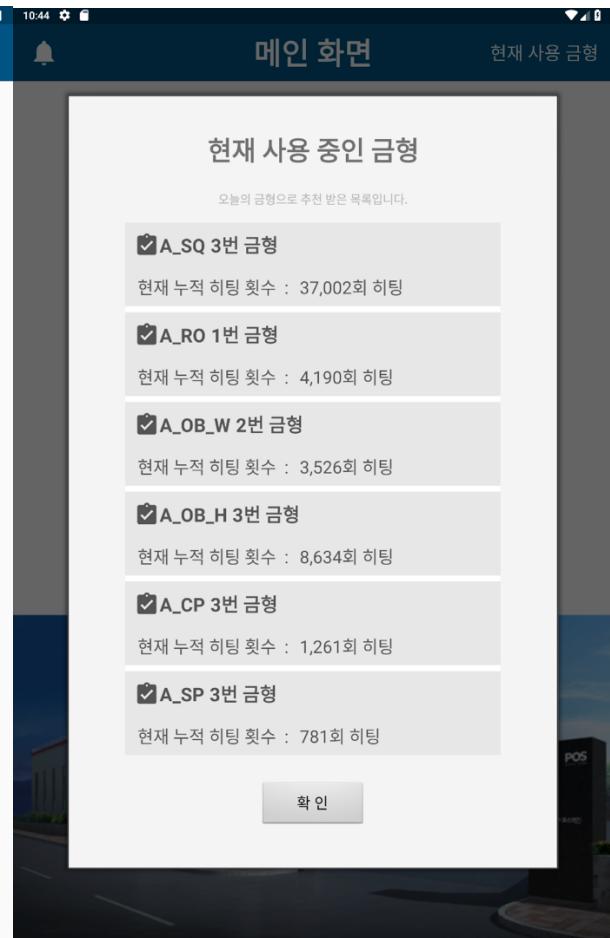
<Figure 35. 앱 메인 화면>

먼저 메인 화면이다. 크게 6 가지 메뉴가 있다.

좌측 상단은 금형 교체 경고 알람이며, 우측 상단은 현재 사용하고 있는 금형을 알려준다.



<Figure 36. 금형 추천(주간)>



<Figure 37. 현재 사용 중 금형 >

금형 추천 화면(주간)은 각 금형마다 2 번째로 편창 횟수가 적은 금형들을 추천한다. 이는 야간 화면과 동일하며 야간에는 가장 작은 편창 횟수의 금형을 추천한다.



<Figure 38. 생산 정보 입력>



<Figure 39. 생산 정보 확인>

생산 정보 입력화면이다. 작업 할 설계도와 생산량, 생산 날짜, 생산 시간, 마지막으로 주간 작업과 야간 작업을 선택한다. 확인 버튼을 누르게 되면, 설계도 별로 필요한 금형이 나오고 판 두께, 총 생산 소요시간이 나오게 된다. 이 정보는 생산 히스토리 메뉴에서 볼 수 있다.

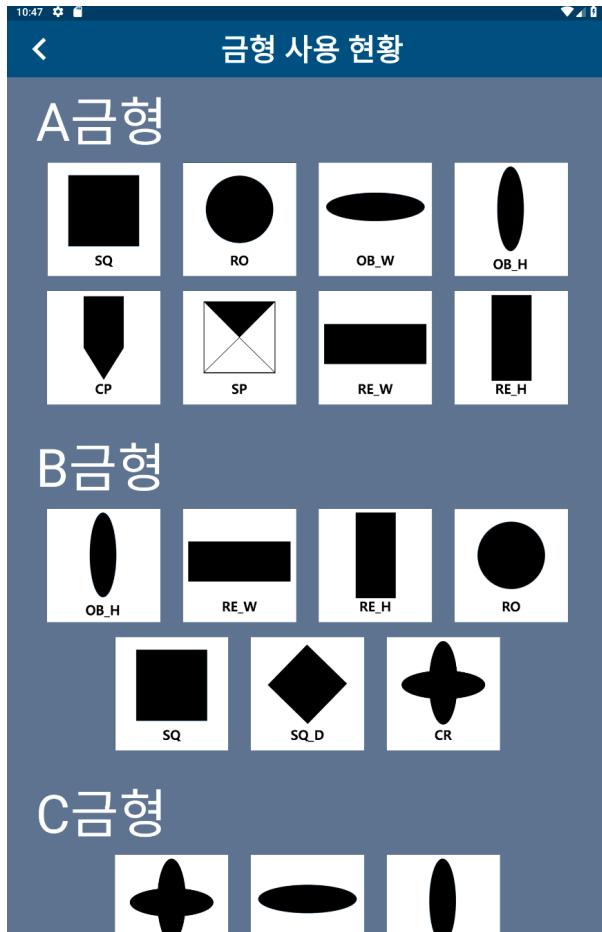


<Figure 40. 설계도 목록>

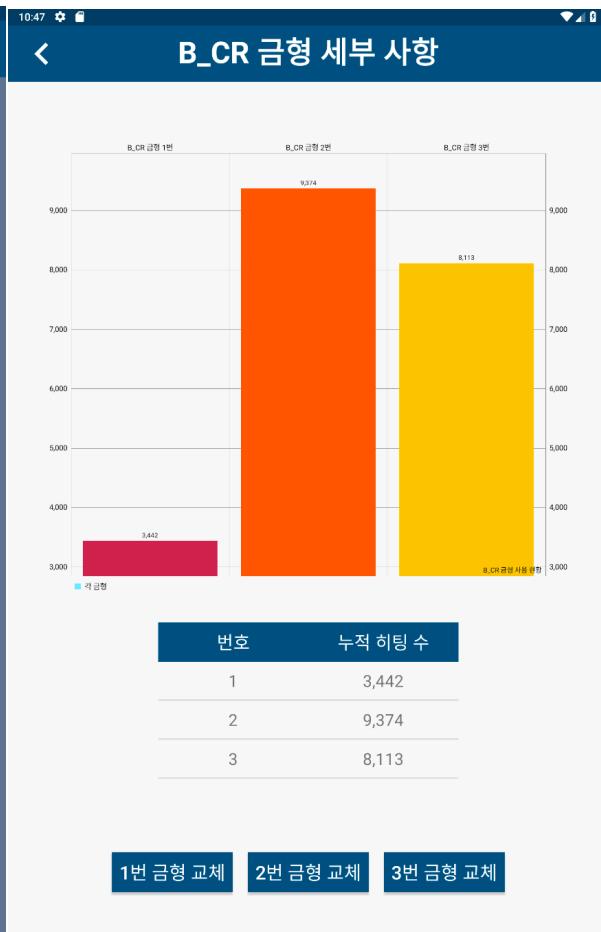
<Figure 41. 설계도 등록>

DataBase에 저장되어 있는 설계도 목록 화면이다.

새로운 설계도를 등록할 수 있도록 설계도 등록 기능도 설계하였다.

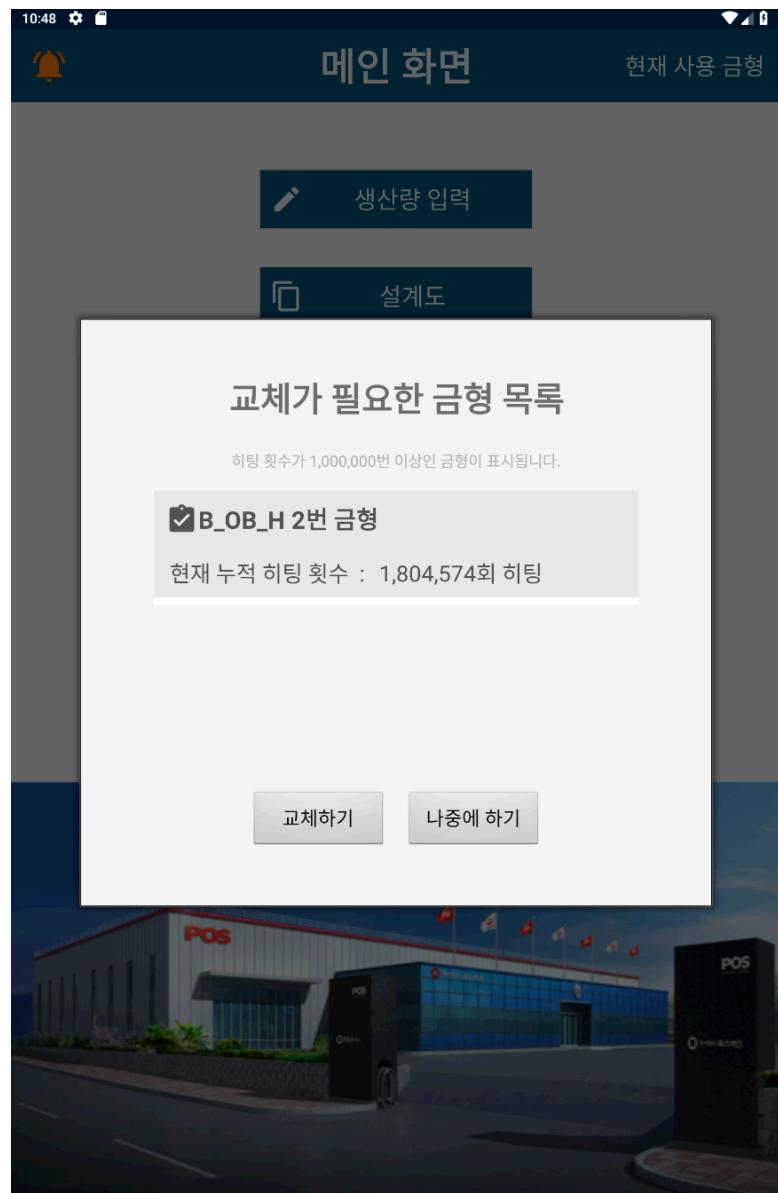


<Figure 42. 금형 사용현황 목록>



<Figure 43. 금형 사용현황 세부>

금형 사용현황 목록화면이다. 각 금형의 그림을 누르게 되면 <Figure 43> 과 같이 편칭의 횟수 여부를 그래프로 나타내 준다. 또한 이 화면에서 편칭 한도가 가까워지면 바로 교체할 수 있게 교체 기능도 추가하였다.



<Figure 44. 금형 교체 경고>

금형 교체 경고 화면이다. 편팅 한도에 가까운 금형을 교체하라는 알림을 보낸다.



<Figure 45. 금형 파손 히스토리>

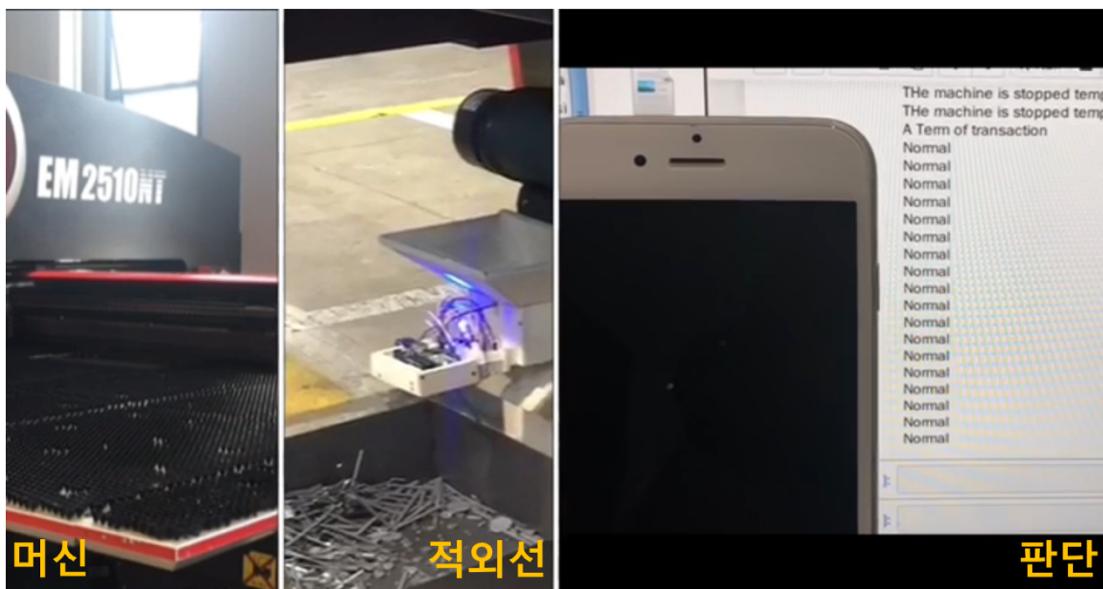
<Figure 46. 금형 파손 등록>

금형 파손 히스토리 화면이다. 지금까지 파손된 금형의 종류와 파손됐을 때 편팅 횟수의 정보를 알려준다.

또한 금형이 파손했을 때, 따로 등록할 수 있도록 금형 파손 등록 기능을 만들었다.

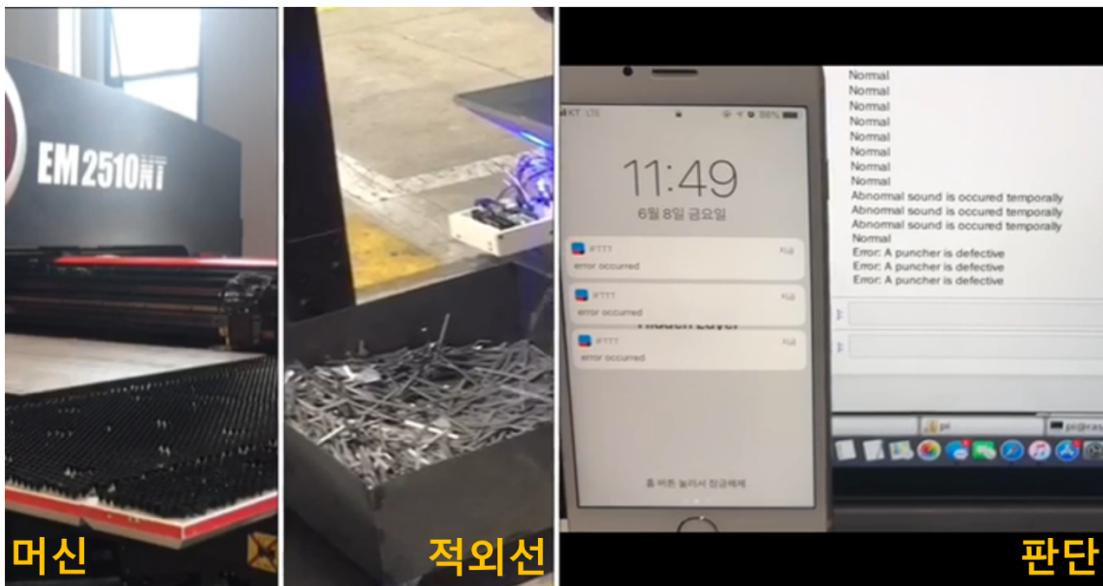
## 4.2 Error Catcher System

다음은 실제 공장에서 검증한 시스템 사진이다.



<Figure 47. 정상 환경일 때>

정상 환경일때에는 판단 화면에서 Normal로 정상이라 판단하며, 알람을 보내지 않는다.



<Figure 48. 비정상 환경일 때>

비정상 환경일때에는 판단 화면에서 “Abnormal sound is occurred temporarily”로 비정상이라 판단하며, 이 판단이 4 번 축적될 시, Error 로 판단하며 핸드폰으로 알람을 보낸다.

## 5. 프로젝트 기대 효과

### 5.1 개발환경

DB	MySQL
Deep learning	Keras , Tensor Flow
Programming language	Python , Java
Arduino	Arduino IDE
App 개발	Android Studio

<Table 6. 개발환경>

### 5.2 프로젝트 기대 효과

#### 1. (주)포스 벽진 자동화 시스템 개선

- 금형 추천 시스템을 구현하여 금형 파손을 예측할 수 있다.
- 추천 알고리즘을 통해 보다 더 효율적으로 금형을 사용할 수 있다.
- 적외선 인식을 사용하여 작업자에게 scrap 이 떨어지는 유무를 시각적으로 보여주어 편리성을 제공하였다.
- 에러 알람 어플을 통해 무인 환경에도 생산 가능성을 제공하였다.

#### 2. 음향 인식을 통한 Defect 검출

- 요즘 큰 이슈가 되고 있는 Machine Learning 을 통해 비정상을 감지 할 수 있는 시스템을 구축하였다.

### 5.3 향후 과제 및 최종 결과물 활용 계획

번호	활용계획	비고
1	시스템 도입	실제 공장에 시스템 도입
2	논문 작성	4 차 산업혁명에 따른 공장 개선을 주제로 논문 작성 및 제출

<Table 7. 최종 결과물 활용 계획>

## 6. 프로젝트 수행 후기

### 이승진

2017년 겨울 주제선정부터 기업 컨택까지 너무나 막연하고 막막했던 캡스톤디자인 프로젝트가 약 6개월간의 노력끝에 막을 내렸다. 우여곡절이 많았지만 팀원들이 힘들고 예민한 시기에도 서로 이끌어주면서 끈끈한 팀워크를 유지할 수 있어서 감사하게 생각한다. 이번 프로젝트를 통해 인간관계의 중요성과 동시에 어려움을 겪을 수 있었고, 배울 수 있었다. 또한 이번 프로젝트를 통해 자발적으로 논문과 서적을 통해 필요한 부분에 대해 스터디하고 연구하는 방법을 충분히 습득할 수 있었다. 특히, 최근 큰 이슈가 되고 있는 빅데이터, 머신 러닝이라는 주제를 다룰 수 있어서 큰 의미가 있다고 생각한다. 전공분야와는 다소 거리가 있는 분야라 생소한 부분이 있었지만, 지금까지 수강한 자료구조와 알고리즘, 데이터 마이닝, 데이터베이스 등의 프로그래밍과 데이터 처리 강의들이 기본을 다지는데 큰 도움이 되었다. 이렇게 캡스톤 디자인이라는 개념에서 정말 의미 있는 프로젝트였던 것 같다. 물론 대학생활 중 가장 힘든 시기를 꼽자면 이번 캡스톤디자인 프로젝트라고 할 수 있지만, 그만큼 성장할 수 있었고 의미 있는 대학 생활의 성과를 이룰 수 있었다고 생각한다.

### 강인선

이번 한 학기는 정말로 내 인생에 인상깊었던 6개월로 기억에 남을 것 같다. 6개월 동안 4명의 팀원이 모두가 한마음으로 뜻을 모아 열심히 노력했던 것들과 결국, 열심히 노력한것으로 인해 얻어 냈던 성과들을 성취해낸 성취감 이런 것들이 정말 오래도록 남을 것 같고, 인생에 굉장히 도움이 많이 될 것 같다. 6개월 동안 같이 고생했던 팀원들에게 감사하고, 교수님들도 모두 수고하셨다는 말씀을 드리고 싶습니다.

## 오익준

졸업프로젝트는 많은 방면에서 나에게 도움이 된 것 같다. 내가 할 수 없다고 생각했던 것들이 생각만큼 어렵지 않았다는 것을 배웠다. 인간관계에 대해서도 많은 것을 배울 수 있었던 계기가 되었다. 팀원들과 함께, 할 수 있다는 마음가짐으로 졸업프로젝트에 임해서 좋은 결과를 얻을 수 있었던 것 같다.

팀원들 모두 정말 수고 많았고 앞으로도 어떤 일이든 잘 해내길 바랍니다.

## 이정하

겨울방학이 시작하자마자 시작한 프로젝트인 만큼 시간과 에너지를 다 투자했다. 처음 주제를 선정하기까지 막연하고 힘들었지만, 한 단계 한 단계 나아갈 수록 성취감도 맛보았고 문제를 해결하는 능력도 키울 수 있었다. 무엇보다도 팀이라는 걸 많이 배웠던 6개월이었다. 팀원 모두가 함께 열심히 하고, 서로 격려하는 배려심과 이해심이 있을 때 비로소 팀의 능력을 얻을 수 있으며, 끝까지 갈 수 있다는 것을 알게 되었다. 선배님들과 교수님들 말씀처럼 이 프로젝트를 통해 배운 점이 많았고, 또 이런 값진 경험을 다른 학교 학생들과 달리 우리는 대학생 때 배울 수 있어서 좋았다.

우리 과의 꽃은 졸업 프로젝트라는 말이 있는데 고생한 만큼 그 꽃의 열매를 잘 얻게 되어서 매우 뿌듯하고 감사하다. 팀원들 모두 너무 고생했고 수고하셨습니다.

## 7. 참고문헌

NO	내용
1	동전에 반응하는 저금통(2014), <a href="https://kocoafab.cc/make/view/173">https://kocoafab.cc/make/view/173</a>
2	라즈베리파이 쿡북, 사이먼 몽크, 박경욱, 백운혁, 유시형 한빛 미디어 (2015)
3	모두의 라즈베리파이 with Python, 이시이 모루나, 에사키 노리히데, 길벗 (2016)
4	모두의 라즈베리파이 with Python, 이시이 모루나, 에사키 노리히데, 길벗 (2016)
5	아두이노 강좌(2017), <a href="https://blog.naver.com/mchoi0602/220918629042">https://blog.naver.com/mchoi0602/220918629042</a>
6	음악신호 머신러닝 초심자를 위한 가이드, <a href="http://keunwoochoi.blogspot.kr">http://keunwoochoi.blogspot.kr</a>
7	코딩셰프의 3분 딥러닝 케라스맛, 김성진 지음 (2018)
8	파이썬 라이브러리를 활용한 머신러닝, 안드레아스 뮐러, 세라 가이도 한빛 아카데미 (2017)
9	하율의 코딩채널 (2017)
10	Arjovsky, Martin, Chintala, Soumith, and Bottou, Le' on. Wasserstein GAN. In ICML, 2017.
11	hris Donahue, Julian McAuley , Miller Puckette. Synthesizing Audio with Generative Adversarial Networks. 2018
12	Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville. Improved Training of Wasserstein GANs
13	Unsupervised classification of heart sound recordings2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference Tsai, Wei-Ho
14	<a href="https://ksy3241blog.wordpress.com/2016/06/17/sudo-apt-get-update-에러/">https://ksy3241blog.wordpress.com/2016/06/17/sudo-apt-get-update-에러/</a>
15	<a href="http://scikit-learn.org/stable/developers/advanced_installation.html">http://scikit-learn.org/stable/developers/advanced_installation.html</a>
16	<a href="http://www.amada.fr/media/user/doc_655.pdf">http://www.amada.fr/media/user/doc_655.pdf</a> , Amada puncher
17	<a href="http://www.knight-of-pi.org/installing-jupyter-on-a-raspberry-pi-for-notebooks-debugging-and-data-analysis/">http://www.knight-of-pi.org/installing-jupyter-on-a-raspberry-pi-for-notebooks-debugging-and-data-analysis/</a>
18	<a href="http://www.ktman.pe.kr/RaspberryPi/59483">http://www.ktman.pe.kr/RaspberryPi/59483</a>
19	<a href="http://www.instructables.com/id/Installing-Keras-on-Raspberry-Pi-3/">http://www.instructables.com/id/Installing-Keras-on-Raspberry-Pi-3/</a>
20	<a href="https://www.raspberrypi.org/forums/viewtopic.php?t=124936">https://www.raspberrypi.org/forums/viewtopic.php?t=124936</a>
21	<a href="https://www.raspberrypi.org/forums/viewtopic.php?t=25173">https://www.raspberrypi.org/forums/viewtopic.php?t=25173</a>
22	<a href="http://pinkwink.kr/1100">http://pinkwink.kr/1100</a>

## [별 첨]



- 안전 재고 : 80EA
- 도색 공정 후 생산 LT : 20min  
(조립-포장)
- 조립 공정 LT : 15min
- 긴급 주문 횟수 : 10times / month
- 거래처의 납기 기한 :
  - 표준 제품 (4days)
  - 설계 필요 제품 (7days)
  - 비표준 제품 및 외주 진행 제품 (10days)
- 월 평균 판매량 : 700EA
- 일 평균 생산대수 : 30EA
- 일 평균 생산 금액 : 3,000,000 (₩)  
(RACK 제조비용 평균 : 100,000₩/EA)

<Figure 49. 회사 조감도>

기업명	(주) 포스-벽진
기업 형태	중소기업
소재지	경기도 이천시 신둔면 원적로 419-56
사업 분야	통신용 금속 케비넷 RACK /통신장비 제조업체
종업원 수	23 명
연 매출액	72 억원 (2018.01 기준)
자본금	10 억원 (2017.09 기준)
주요 거래처	통신사 (LG / KT / SK), 삼성 등

<Table 8. (주)포스 벽진 기업 정보 표>



<Figure 50. 기계 종류>



<Figure 51. 주요 제품 RACK>

생  
산  
시  
설

공정 순서	사 진	내 용
1	-	<p><b>&lt;고객 주문&gt;</b> 고객이 영업부와 함께 RACK 을 설계하고 주문을 한다.</p>
2		<p><b>&lt;원자재(철판) 입고&gt;</b> 포스코 &amp; 현대제철에서 규격에 맞는 철판을 입고한다.</p>
3	-	<p><b>&lt;입고 검사&gt;</b> 맞는 철판이 입고 되었는지, 불량은 없는지 확인하는 절차를 진행한다.</p>
4		<p><b>&lt;Punching 공정&gt;</b> 구멍을 뚫는 Punching 공정을 설계도면에 따라 진행한다.</p>
5		<p><b>&lt;절곡 공정&gt;</b> Punching 공정이 완료된 철판은 절곡 공정 라인에 들어가게 된다.</p>
6		<p><b>&lt;용접&gt;</b> 절곡이 완료된 철판들은 용접공정을 통해 기본 부품으로 완성된다.</p>
7	-	<p><b>&lt;공정 검사&gt;</b> 모든 공정들이 불량없이 진행되었는지 검사</p>
8		<p><b>&lt;도색 (외주)&gt;</b> 가공 공정이 완료된 제품은 외주로 보내 도색 과정을 거친다.</p>
9		<p><b>&lt;조립&gt;</b> 도색이 완료된 부품은 설계도면에 맞게 부품을 조립하여 완제품을 만든다.</p>
10		<p><b>&lt;포장&gt;</b> 완제품은 비닐로 포장한다.</p>
11	-	<p><b>&lt;보관 및 출하&gt;</b> 출하돼야 할 제품들은 박스 포장을 한 뒤, 완제품 재고에 보관한다.</p>

<Table 9. 공정 순서도>

- 금형 종류



<Figure 52. 금형>

금형종류	금형모양			
A	SQ	OB_W	RE_W	
	RO	OB_H	RE_H	
	CP	SP		
B	OB_H	RO	CR	
	RE_W	SQ		
	RE_H	SQ_D		
C	OB_W	OB_H	CR	
	SP	RR		
D	RE_W	RE_H		
E	RE_W	RE_H		
G	CR	RE_W		

<Table 10. 금형 모양>

평균 근무 일수: 22 days / 1month

평균 야간 일수 : 15 days / 1month

평균 에러 횟수 : 2.5 times / 1month

금형 재고 : 2~3 개 안전 재고 보유

### - 철판 종류

	1T	1.2T	1.6T	2T
용도	통풍구	문	바닥	기둥
설비 종류	반자동 머신		자동 머신	
가격(₩)	23,000	26,000	31,000	36,400
무게(kg)	29.8	32.5	37.3	46.7

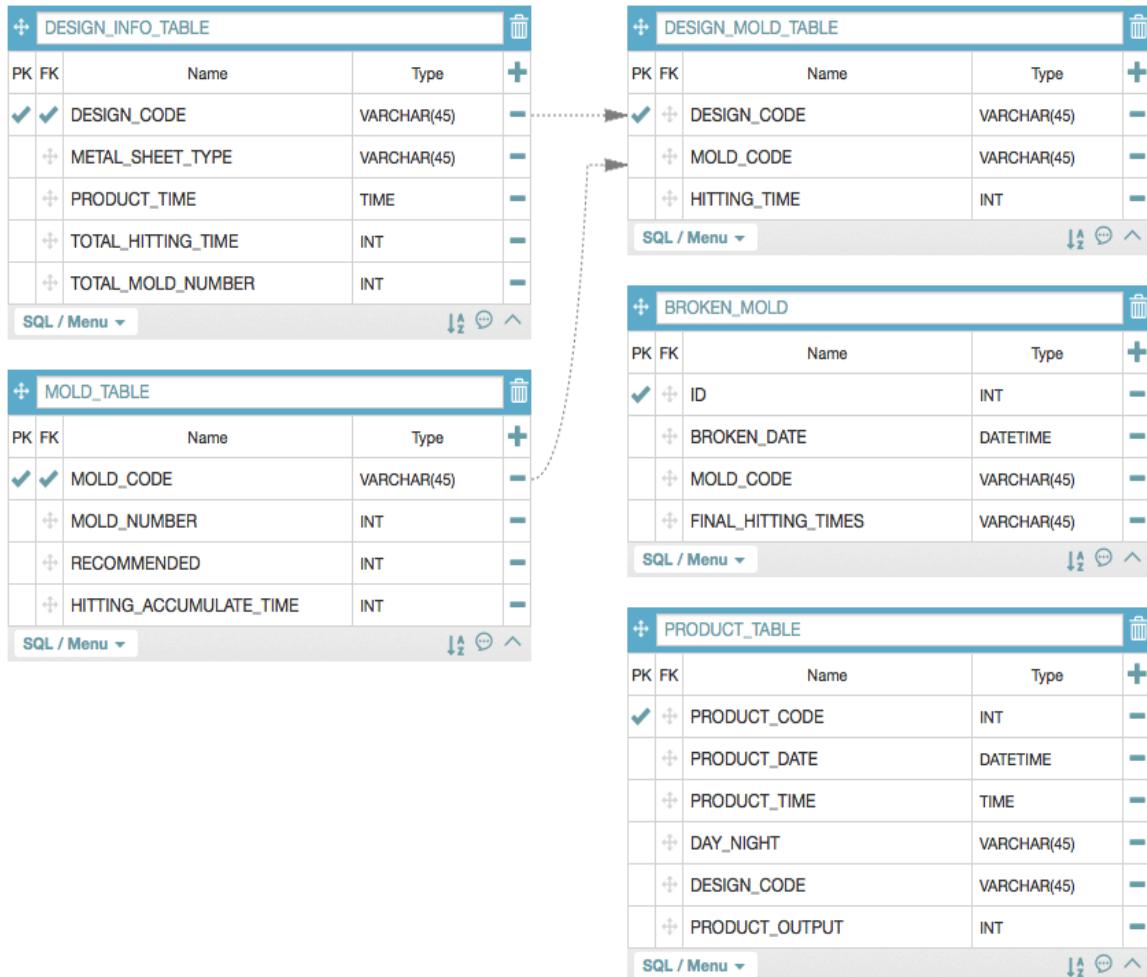
<Table 11. 철판 종류 표>

- Generate Adversarial Network (GAN) Objective Function

→ Discriminator should maximize

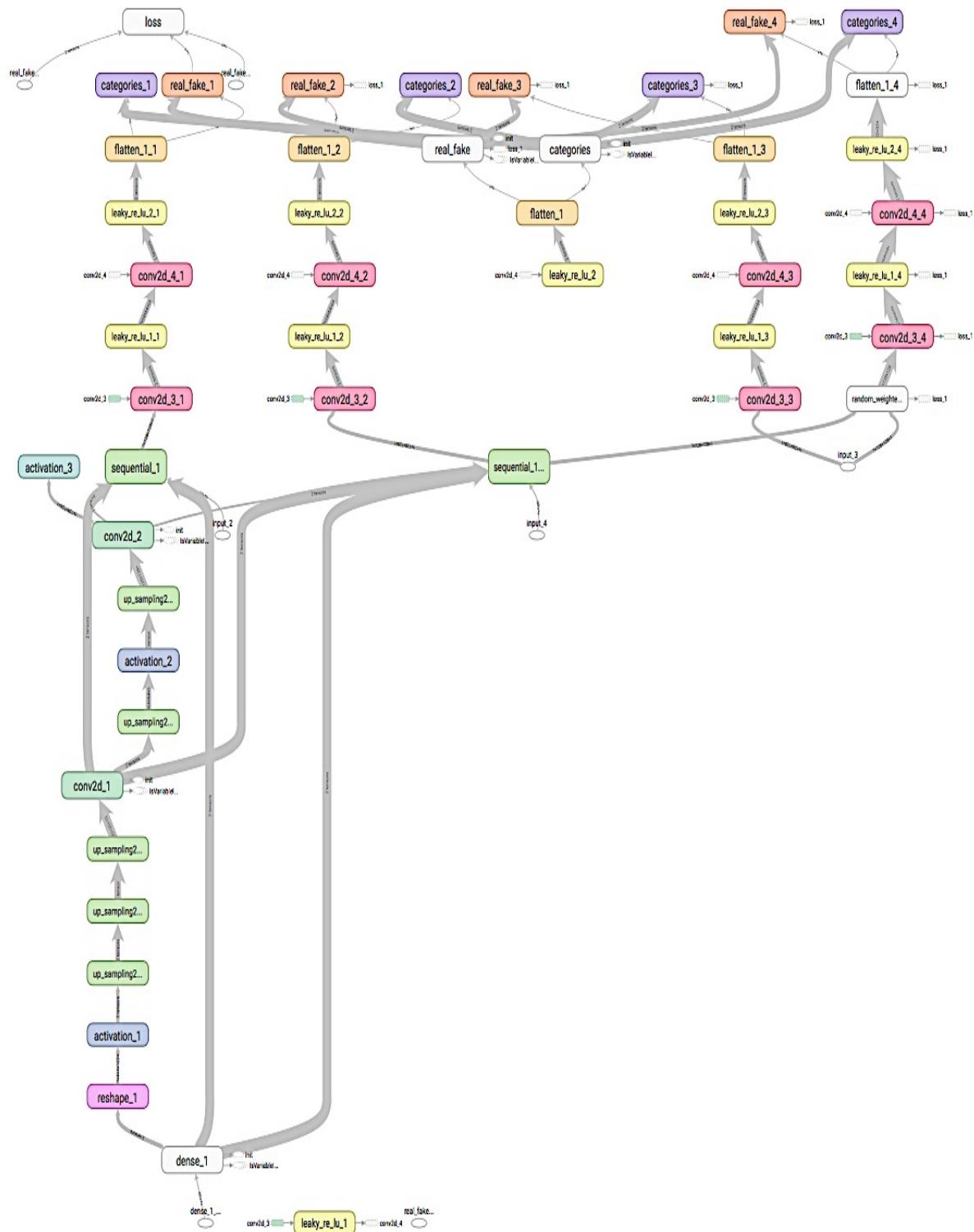
→ Generator should minimize

- Error Prevention System ERD



<Figure 53. 시스템 ERD>

## - Hidden Layer 구조도



<Figure 54. Hidden layer 구조도>