

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. 중간발표 이후 진행 방향	4
2.1 중간발표 지적 사항에 대한 수정 및 보완 내용	4
2.2 최종 산출물	5
2.3 최종적으로 평가를 받고자 하는 내용	5
3. 프로젝트 소개	6
3.1 프로젝트 주제 및 필요성	6
4. 프로젝트 내용	7
4.1 데이터 수집	7
4.2 구현 방안	9
4.3 개발환경	25
5. 프로젝트 수행 계획	25
5.1 업무 분담	25
5.2 프로젝트 일정 및 회의 일정	25
6. 참고자료	26
7. 별첨	27

1. 프로젝트 요약문

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

2. 중간발표 이후 진행 방향

2.1 중간 발표 지적 사항에 대한 수정 및 보완 내용

	지적 사항	수정 및 보완
정확도	목표 정확도를 명시해야 할 것	실험을 통해 목표 정확도를 설정할 예정
	Noise 를 제거하는 방법에 대해 명확히 기술해야 할 것	1 차적으로 noise 를 포함한 training 을 통해 noise 를 오류로 인식하지 않고 적절한 구간을 설정하여 의사결정을 함으로써 짧게 일어나는 noise 는 의사결정에 큰 영향을 끼치지 않게 할 것이다.
	Type1, Type2 Error 를 명시해야 할 것	1 종 오류는 알람을 보낼 시 음향 파일을 함께 발송하여 작업자가 직접 불량 소리를 들을 수 있게 설계. 2 종 오류는 scrap 이 안 떨어지는 경우에는 물리적 반응이 있으므로 분명히 확인할 수 있지만 한계점이 있고 극복하기 위해 차후 연구한 내용을 반영할 예정
신뢰성	앞으로의 신뢰성 및 정확성 검증 필요	장치 설치 이후 다양한 조건의 실험을 통해 검증을 할 예정
진행사항	금형 추천 시스템의 진행 상태가 미흡	현재 App 구현 중이며, 금형을 추천하는 계산 과정에서 신뢰도를 고려하여 추천하도록 설계할 것

2.2 최종 산출물

2.2.1 Error Catcher (EC) System 구현

- 실시간 음향 인식 판단
 - 실시간 적외선 인식 판단
 - 실시간 음향 인식 & 적외선 인식 App 연동
- ### 2.2.2 금형 추천 시스템 (EP System) 구현
- 금형 사용 현황 & 추천
 - 금형 파손 예측

2.3 최종적으로 평가를 받고자 하는 주요 내용

2.3.1 음향 인식과 적외선 센서에 대한 정확도

- 실시간으로 데이터 분석이 가능한가?
- 음향 인식과 적외선 센서가 서로 상호보완적인가?
- 실제 결과가 기대효과에 부응하였는가?
- 인식 결과가 어느정도 정확한가?

2.3.2 EP 시스템에 대한 실용성

- 작업자 측면에서 번거롭지 않게 시스템이 구현되었는가? (작업자 만족도)
- 금형 추천 알고리즘이 논리적인가?

2.3.3 False Alarm에 대한 대처

- 적절한 false alarm 대처를 반영하였는가?

3. 프로젝트 소개

3.1 프로젝트 주제 및 필요성

3.1.1 프로젝트 주제

- 자동화 설비 개선

1) Error Prevention (EP) System 구축

야간 작업 시 Min. Punching 횟수의 금형을 사용하여 Error를 대비하기 위한 금형 추천 시스템. 총 26 가지의 금형을 각각 3 개씩 유지하여 생산할 때마다 금형을 추천한다.

2) Error Catcher (EC) System 구축

야간 작업 시 발생한 Error를 작업자에게 바로 알리고 대처하기 위한 시스템. 한 가지는 적외선 인식기를 통해 punching 소리가 났을 때 철판 scrap이 떨어지지 않으면 Error라고 판단하여 알람을 보내고 나머지 한 가지는 음향 인식을 통해 training 된 정상 소리가 아닌 비정상 소리라고 판단했을 때 Error라고 판단하여 알람을 보낸다. 위 두 가지를 함께 고려하여 Error를 판단 정확도를 높인다.



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

Error Prevention (EP System)



야간 작업시 발생한 Error를
작업자에게 바로 알리고 대처하기 위한

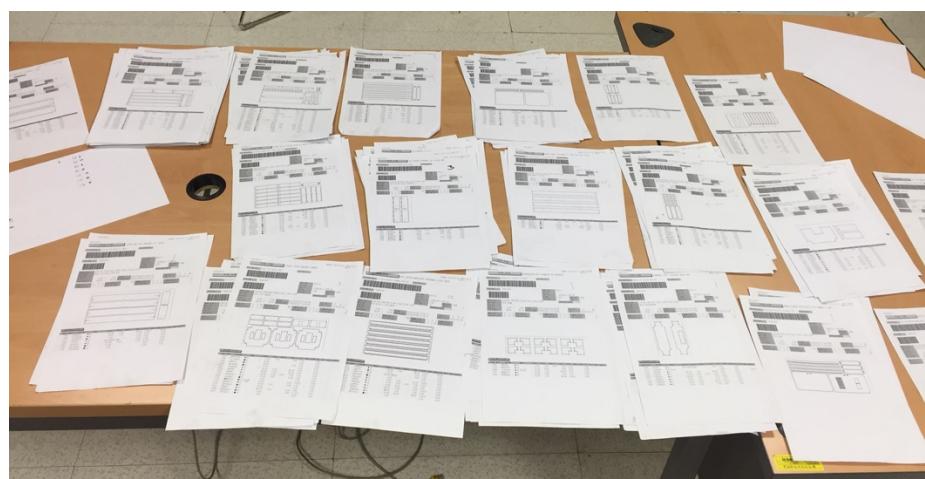
Error Catcher (EC System)

4. 프로젝트 내용

4.1 데이터 수집

4.1.1 EP 시스템

생산할 때마다 금형을 추천하기 위해서, 각 설계도 별 어떤 금형이 쓰이는지 알아야 한다. 따라서 실제 기업에서 사용하고 있는 종이로 된 설계도를 excel 파일로 저장하였다. 설계도 파일 안에는 금형 별 punching 횟수, 총 punching 시간, 사용하는 금형 종류, 사용하는 철판 종류 등이 있다.



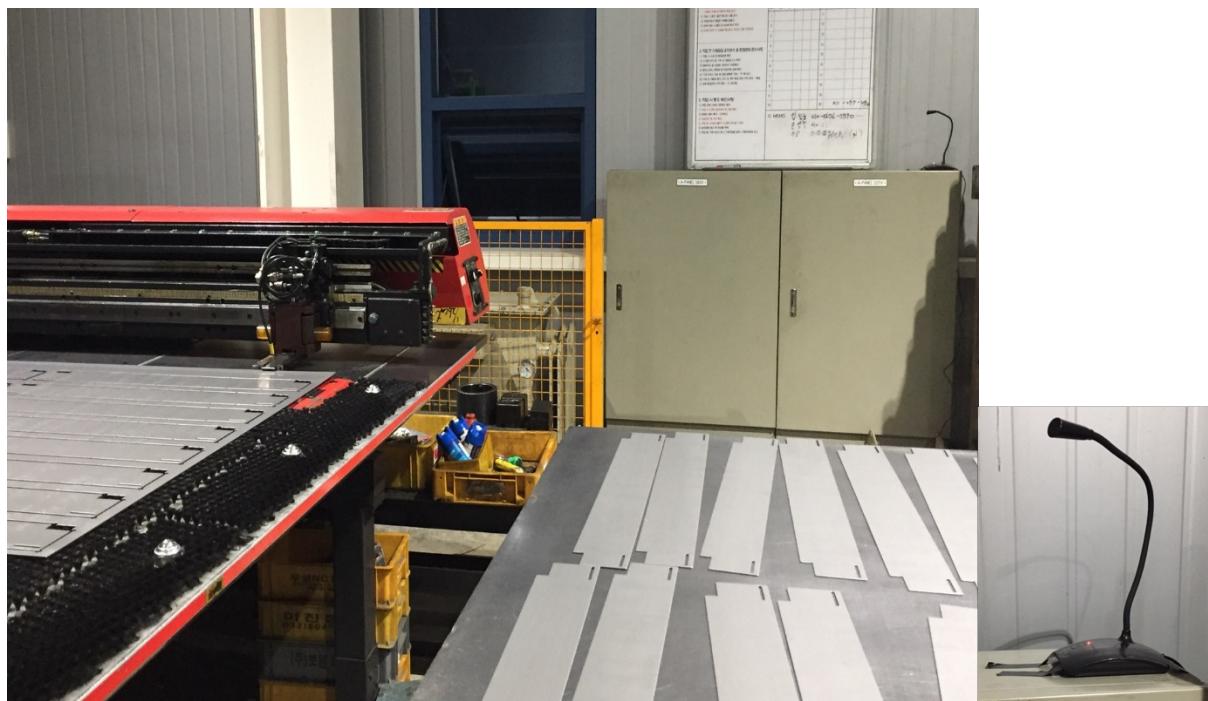
<Figure 1. 실제 사용하는 설계도>

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	
1 DESIGN_CODE	METAL_SHEET_PUNCHING_TA_SQ			A_RO	A_OB_W	A_OB_H	A_CP	A_SP	A_RE_W	A_RE_H	B_OB_H	B_RE_W	B_RE_H	B_RO	B_SQ	B_SQ_D	
2 GUIDE_ANGLE	2T	94	0	48	0	0	0	0	0	0	32	62	0	0	40	0	
3 RING_RUN	1.6T	292	0	56	78	84	0	0	0	0	0	0	0	180	520	0	
4 BASE	2T	256	0	27	0	37	0	0	0	0	0	25	0	168	122	0	
5 TOP_BASE_A	1.6T	379	21	79	24	0	0	0	0	123	0	0	54	92	47	79	0
6 BLANK_PANEL	1.6T	431	0	28	77	26	32	0	288	0	121	0	42	28	27	0	
7 KEYBOARD_MAIN_BODY	1.6T	277	0	20	16	0	32	0	0	167	0	28	0	51	66	0	
8 OIL_1	1.6T	133	0	45	50	0	0	0	0	146	0	0	14	0	78	0	
9 OIL_2	2T	146	0	14	0	0	21	0	0	0	23	114	78	7	29	0	
10 PANEL_TOP	1.6T	78	0	24	0	32	0	0	0	0	0	0	0	0	32	0	
11 SHELF_1	1.6T	40	0	8	0	0	0	0	0	0	0	0	0	0	0	0	
12 PANEL_1140	1.6T	109	0	48	0	22	0	0	0	0	0	0	0	0	0	0	
13 SHELF_2	1.6T	67	0	7	0	0	12	0	0	0	0	0	0	0	48	0	
14 SIDE_FRAME_C	2T	169	0	93	0	66	0	0	0	0	0	62	43	0	140	0	
15 SIDE_FRAME_B	2T	417	226	210	0	283	0	0	0	0	0	84	104	0	179	0	
16 SIDE_FRAME_A	2T	356	210	162	93	121	0	0	0	0	0	0	0	0	218	0	
17 SIDE_FRAME_D	2T	65	0	28	12	0	0	0	0	0	0	0	0	0	18	0	
18 MAIN_FRAME_F	2T	247	92	87	0	0	0	40	134	0	0	0	0	0	0	0	
19 SIDE_FRAME_G	2T	402	0	273	69	0	0	0	0	0	0	0	172	0	293	0	
20 MAIN_FRAME_B	2T	231	190	98	0	0	0	21	0	0	0	0	72	0	0	0	

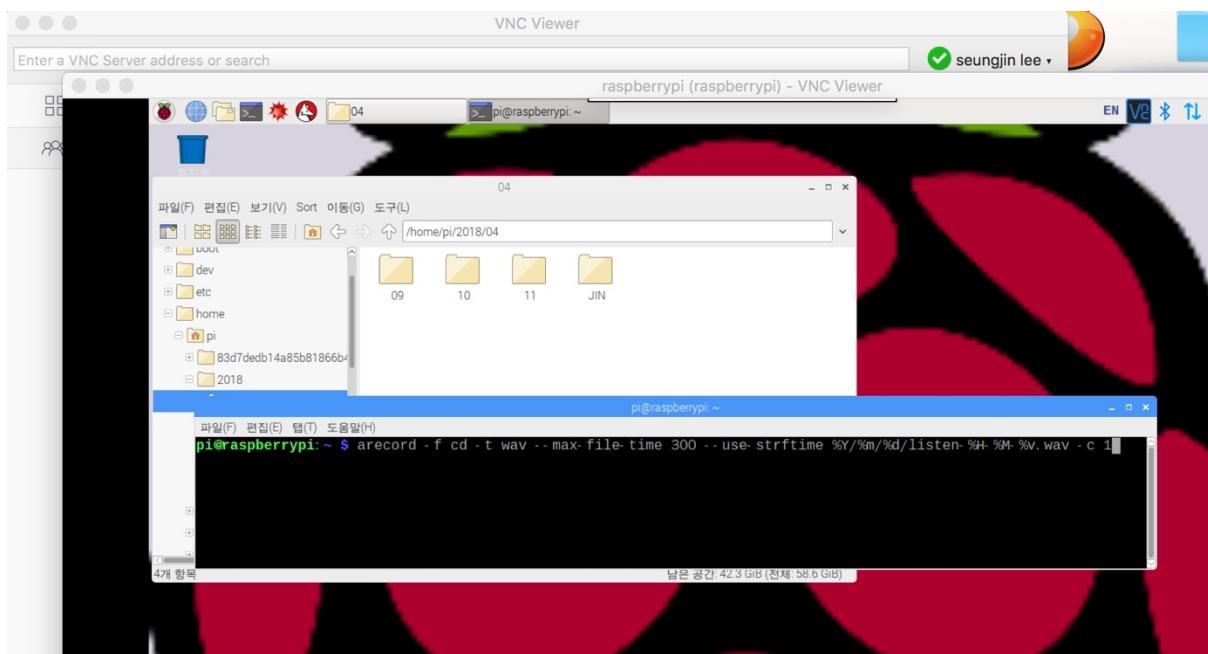
<Figure 2. Excel로 변환한 설계도 data>

4.1.2 EC 시스템

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 수집하고 있다.



<Figure 3. 설비 옆 설치된 마이크>



<Figure 4. VNC 원격화면>

4.2 구현 방안

4.2.1 EP 시스템

금형에 작용하는 요인은 punching 횟수 & 철판 두께 이다.

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

철판 두께는 현재 자동화 설비를 제작하고 있는 아마다에게 문의하여 알아본 결과,
철판 두께에 따라 punching force 가 결정된다는 것을 알 수 있었다.

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 결정된다는
것을 알 수 있다.

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.

금형 추천 계산 식은 다음과 같다.

$$\begin{aligned}\text{남은 punching 수} &= (\text{금형의 punching 한도}) - (\text{사용한 punching 횟수}) \\ &\quad - (\text{생산할 설계도의 punching 횟수} \times \text{생산량})\end{aligned}$$

남은 punching 횟수 < 0 이면, 추천을 하지 않으며

남은 punching 횟수 > 0 일 때, 추천을 한다.

야간 생산일 경우, Min. punching 횟수의 금형을 추천하여 에러 발생확률을 최대한 낮추고, 주간 생산일 경우, 에러에 즉각 대비할 수 있으므로 2 번째로 적은 punching 횟수의 금형을 추천한다.

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.

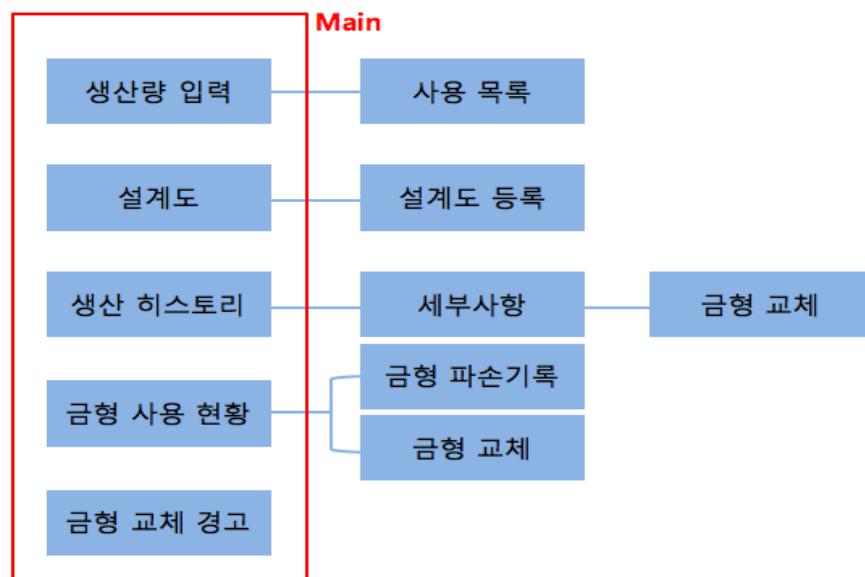
다음 주간 생산에는 2 번째로 punching 횟수가 적은 A_RO_2 번 금형을 추천한다.

그 다음 주간 생산에는 punching 횟수가 2 번째로 적은 A_RO_1 번 금형을 추천한다.

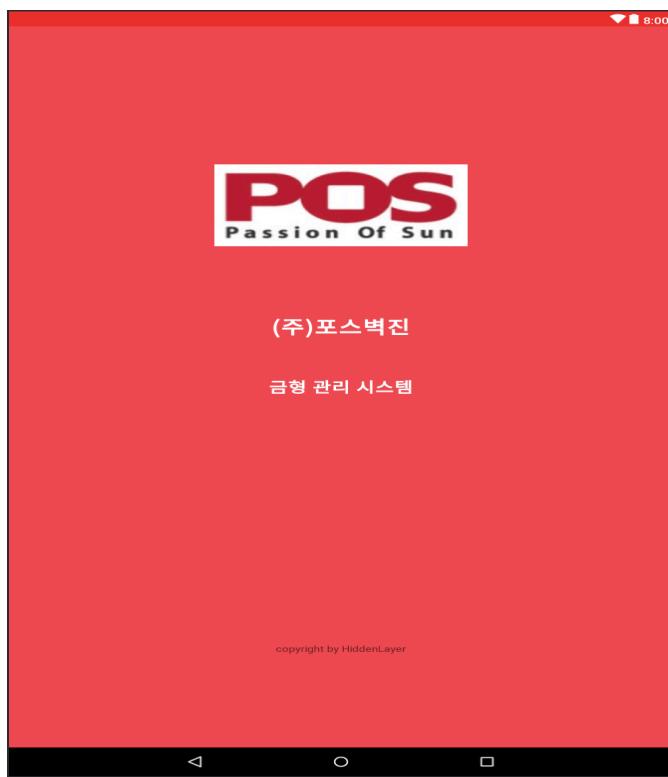
야간 생산에는 Min. punching 횟수인 A_RO_3 번 금형을 추천한다.

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.

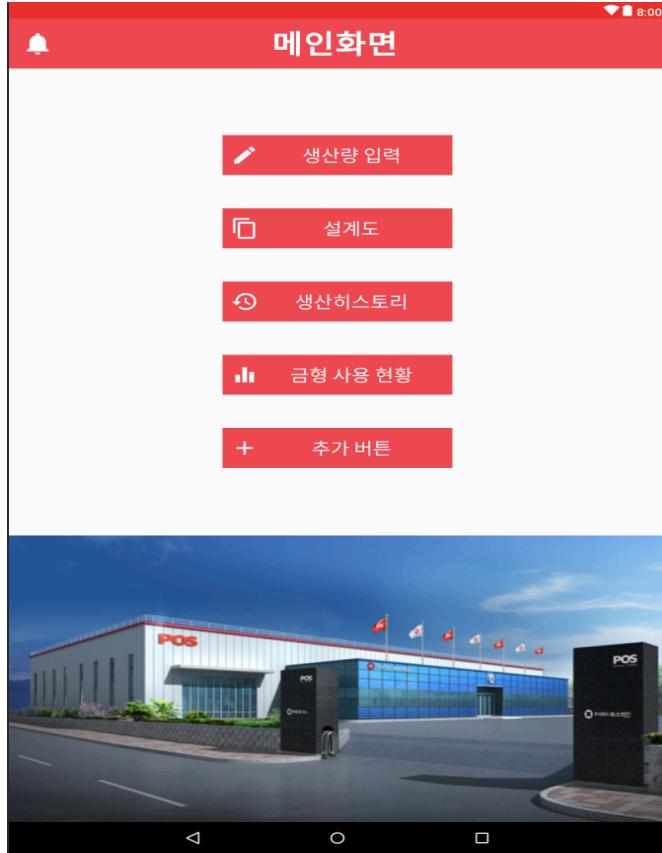
다음은 금형 추천 시스템 (App) 메뉴 구조도 & UI 와 ERD 구조도 이다.



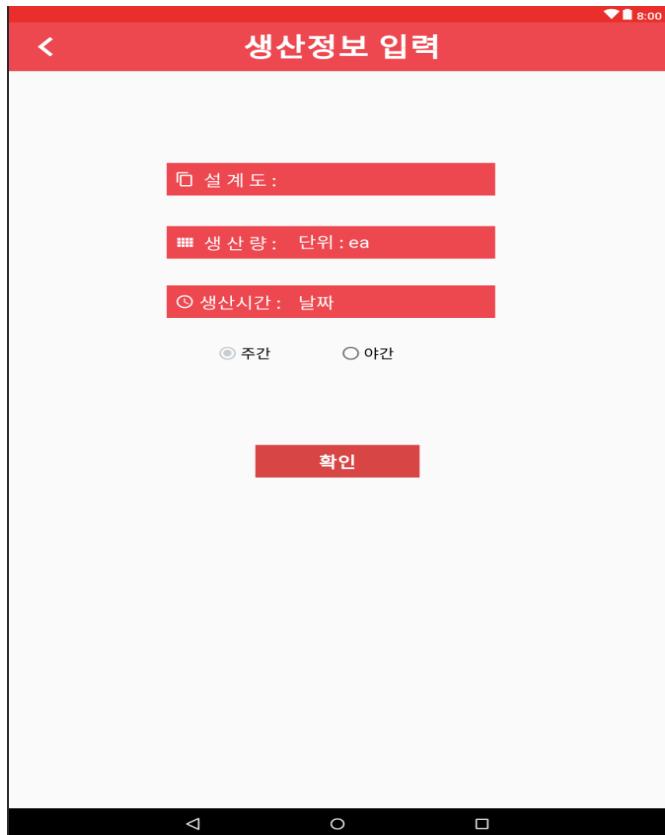
<Figure 5. EP 시스템 메뉴 구조도>



<Figure 6. App Loading 화면>



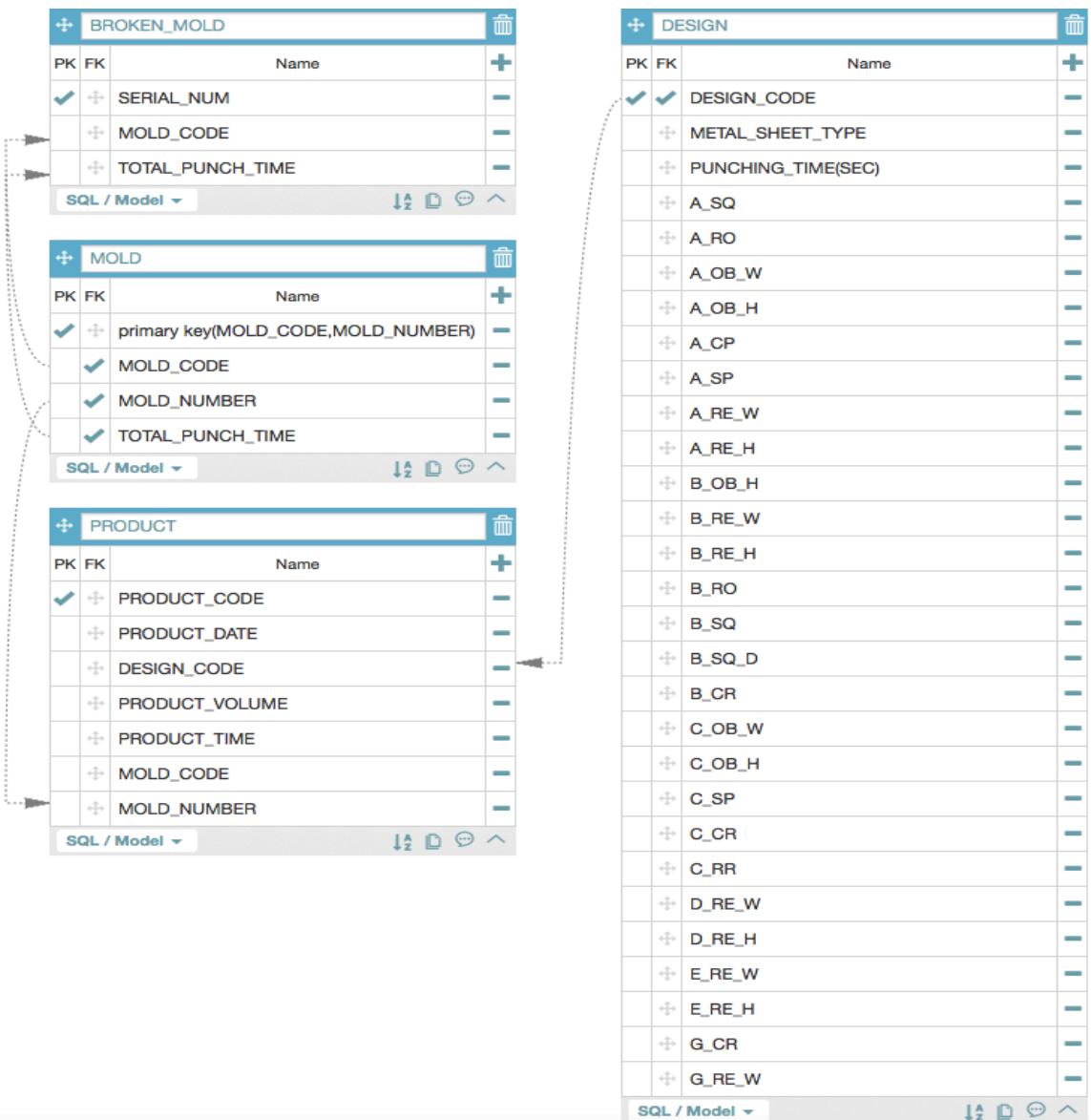
<Figure 7. App Main 화면>



<Figure 8. App 생산 정보 입력 화면>



<Figure 9. App 설계도 목록 화면>



<Figure 10. ERD >

<DESIGN TABLE>

- DESIGN_CODE : 설계도 name
- METAL_SHEET_TYPE : 1.6T / 2T

<PRODUCT TABLE>

- PRODUCT_CODE : auto increment
- PRODUCT_DATE : 생산 날짜
- PRODUCT_VOLUME : 생산량
- PRODUCT_TIME : 주간 / 야간

<MOLD TABLE>

- PRIMARY KEY : MOLD_CODE + MOLD_NUMBER
- MOLD_CODE : 금형 종류(총 26 가지)
- MOLD_NUMBER : 같은 금형의 번호 (1~3)
- TOTAL_PUNCH_TIME : 총 펀칭 횟수

<BROKEN_MOLD TABLE> → 깨진 금형이 생길 시, record 가 추가된다.

4.2.2 EC 시스템

① 적외선 인식

떨어지는 철판 scrap 을 사용하기 위해, 적외선 빔 센서를 사용하여 적외선 인식기를 만들었다.

적외선 빔 센서는 수광 & 발광 센서가 있어서 그 사이에 물체가 있어 센서가 끊기게 되면 물체가 있다는 것을 감지하는 센서이다.

아두이노와 호환이 가능한 초소형 적외선 빔 센서를 사용하여 인식기를 만들었고, 실험 한 결과 철판 scrap 이 빠른 속도로 떨어져 인식하지 못하는 현상이 발생했다. 문제점을 해결하기 위해, 간접적인 방법을 선택하여 인식기 사이에 종이를 설치하였다.

scrap 이 떨어질 때, 종이를 건드려 인식 범위를 벗어나면 아두이노 블루투스 모듈을 통해 어플로 1의 데이터를 전송하고 그렇지 않으면, 0 의 데이터를 전송한다. 어플은 1의 데이터를 받으면 초록 화면을 보여주고, 0의 데이터를 받으면 빨간 화면을 보여준다.



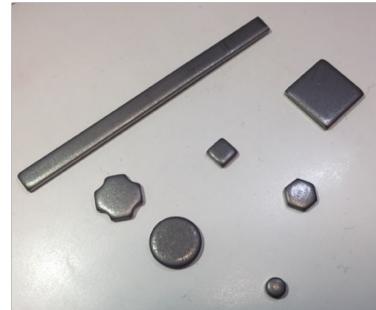
<Figure 11. Scrap 떨어지는 곳>

별도로 적외선 인식기의 성능을 실험하였고 다음과 같다.

- 외부 요인 : 시간, 감지센서 측정 속도, 컨베이어 벨트 속도
- 내부 요인 : 종이 재질, 스크랩 크기 및 개수

실험 내용 : 모든 조건에서 5초 내에 최대 몇개의 스크랩을 인식하는지 실험.

- * 제일 큰 스크랩 : 1.5cm
- * 제일 작은 스크랩 : 0.4cm
- * 제일 긴 스크랩 : 9cm



<Figure 12. 실제 철판 scrap>

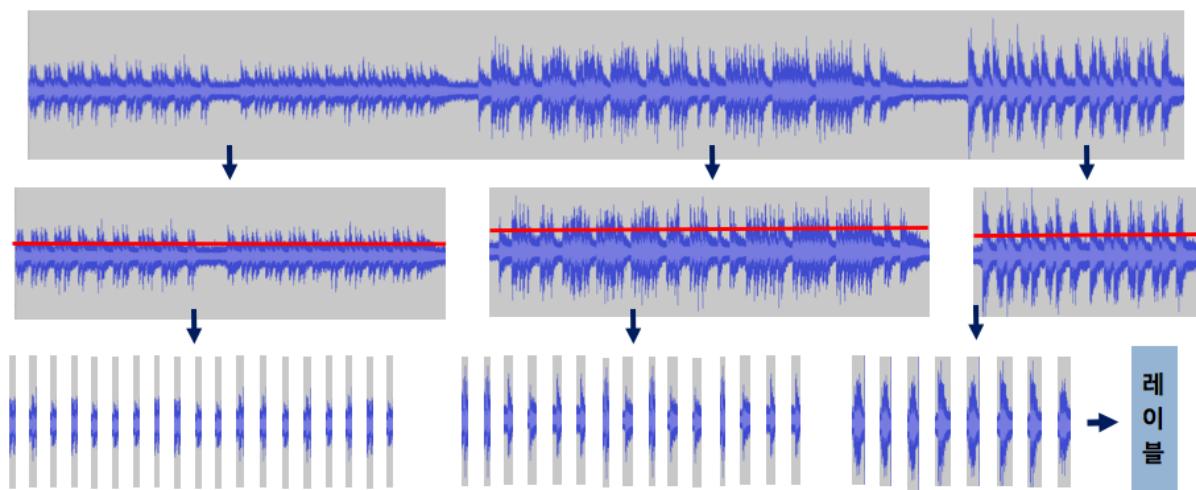
	큰 스크랩	작은 스크랩
코팅지+A4지	O	인식 안됨 (종이위에 쌓임)
코팅지 + 스노우지	O	인식 안됨 (종이위에 쌓임)
누보지(명함)	인식 안됨 (스크랩이 종이위에 쌓임)	
스노우지	10개 스크랩 → 모두 인식 15개 스크랩 → 모두 인식 20개 스크랩 → 모두 인식 22개 스크랩 → 모두 인식 23개 스크랩→1개 인식 불가 <i>Limit : 1개 / 0.22초 인식</i>	10개 스크랩 → 모두 인식 15개 스크랩 → 모두 인식 17개 스크랩 → 모두 인식 18개 스크랩→1개 인식 불가 <i>Limit : 1개 / 0.29초</i>
A4지	10개 스크랩 → 모두 인식 12개 스크랩 → 모두 인식 15개 스크랩 → 모두 인식 17개 스크랩 → 모두 인식 18개 스크랩→1개 인식 불가 <i>Limit : 1개 / 0.29초 인식</i>	10개 스크랩 → 모두 인식 15개 스크랩 → 모두 인식 16개 스크랩→1개 인식 불가 <i>Limit : 1개 / 0.3초</i>

<Table 1. 적외선 인식기 실험>

② 음향 인식

1) 데이터 전처리

녹음된 punching 소리 데이터는 금형 별로 수동으로 편집하여 저장한다. 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 레이블링 되며 training 과정으로 가게 된다.



<Figure 13. 데이터 전처리 과정>

2) 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.

3) 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. 다시 또 Generator 가 그 전보다 실제 이미지와 비슷한 이미지를 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.

다음은 지금까지 training 한 현황이다.

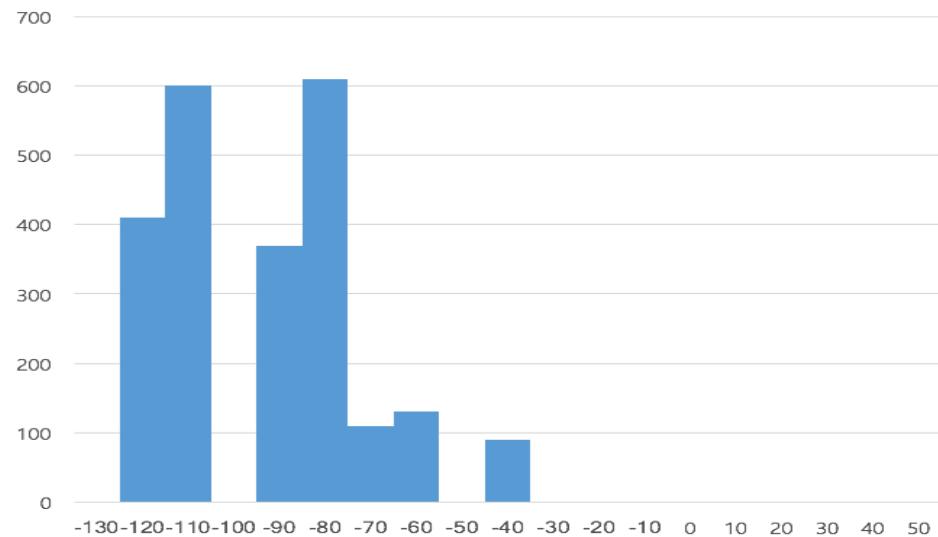
502-1	502-2	502-3	502-4	GPU
Training 기간 : 7일 정확도 : 20% 아래 Epoch : 900 Hidden layer : 3 Batch : 64 → 폐기	Training 기간 : 8일 정확도 : 대략 50% Epoch : 970 Hidden layer : 3 Batch : 64 → 아직 진행중	Training 기간 : 7일 정확도 : 30% 아래 Epoch : 800 Hidden layer : 4 Batch : 64 → 폐기	Training 기간 : 8일 정확도 : 대략 85% Epoch : 1500 Hidden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

<Table 2. Training 현황>

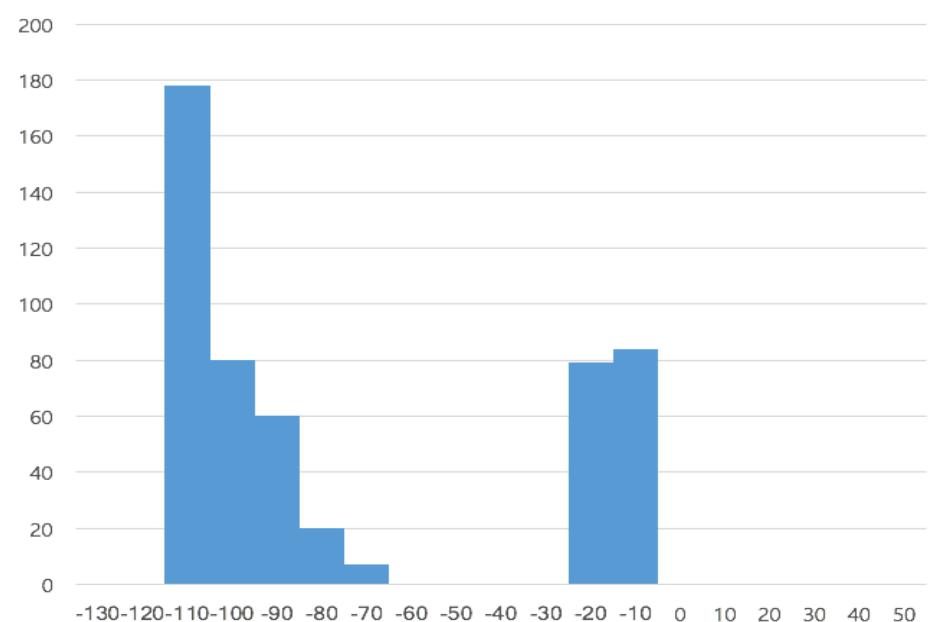
공학과 502 호실에서 4 대의 컴퓨터로 training 하고 있으며, 별도로 GPU 도 함께 training 중에 있다.

지금까지 가장 정확도가 높은 Case 는 502-4 model 이다.

먼저 정확도가 50%인 502-2 model 이다.

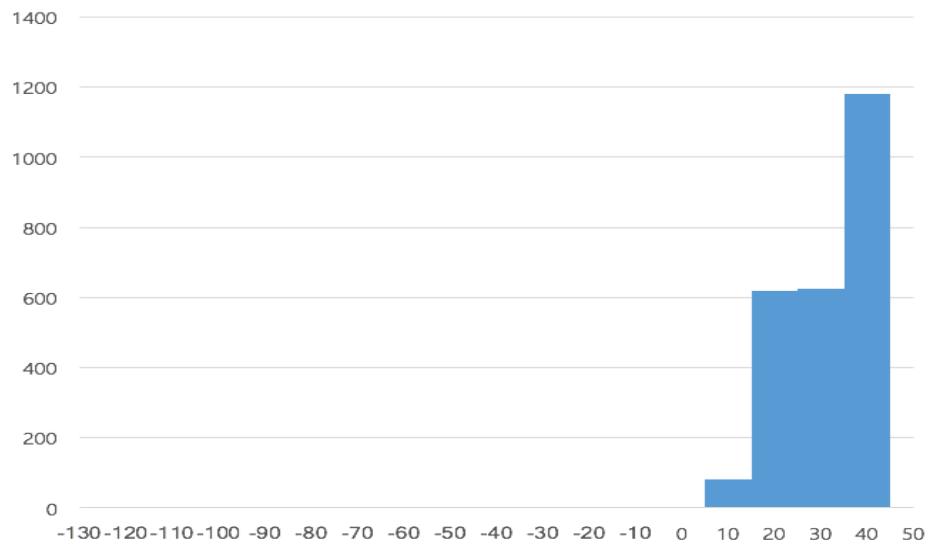


→ Epoch 50 일 때, 정상 데이터 Output

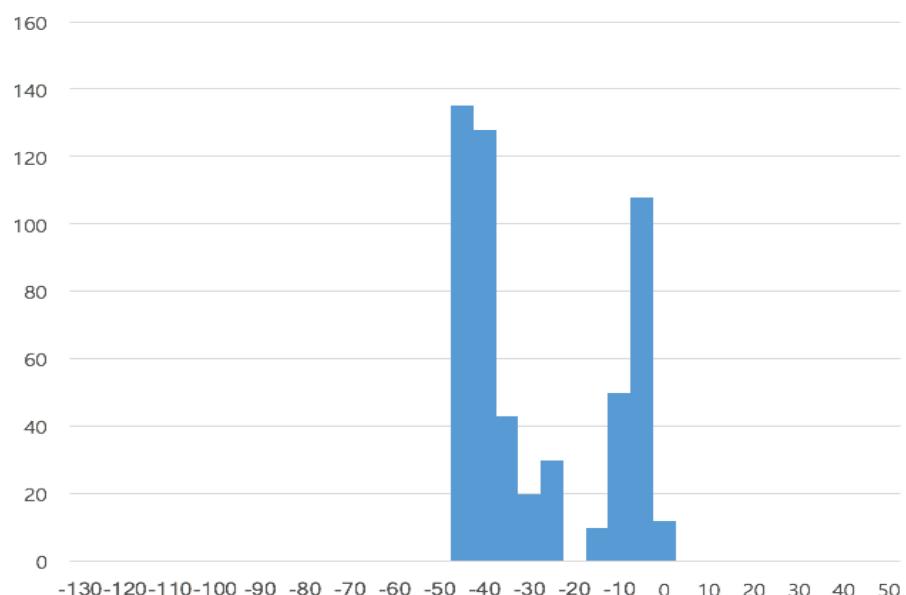


→ Epoch 50 일 때, 비정상 데이터 Output

502-2 model 은 Epoch 가 50 일 때, 정상과 비정상을 완벽하게 구분하지 못함.



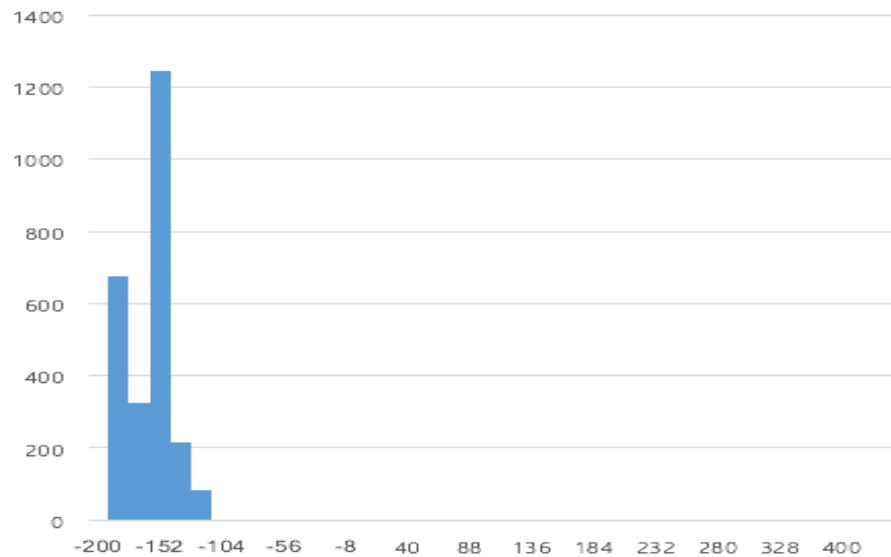
→ Epoch 950 일 때, 정상 데이터 Output



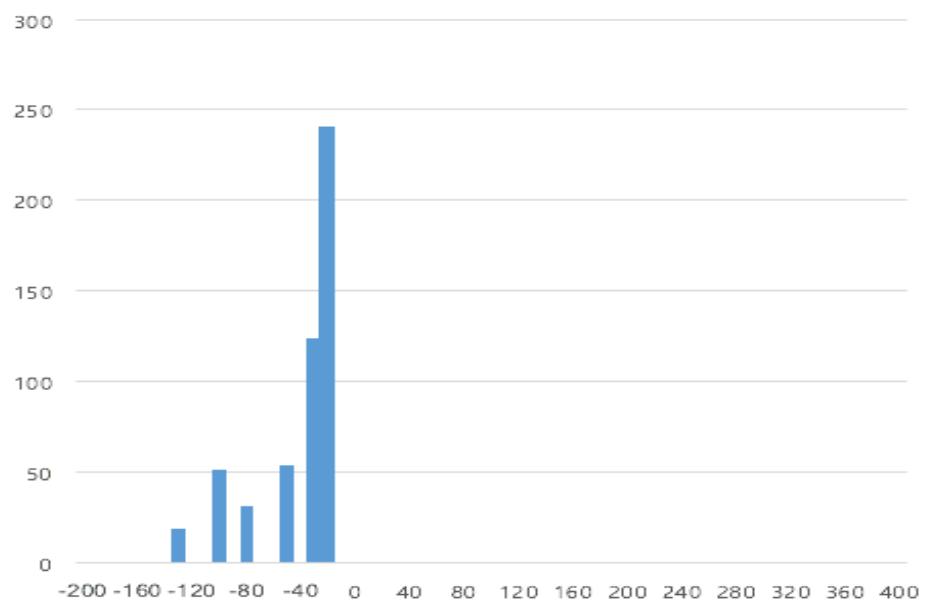
→ Epoch 950 일 때, 비정상 데이터 Output

Epoch 가 950 일 때는 50 일 때보다 정상과 비정상을 정확히 구분함.

다음은 정확도가 85%인 502-4 model 이다.

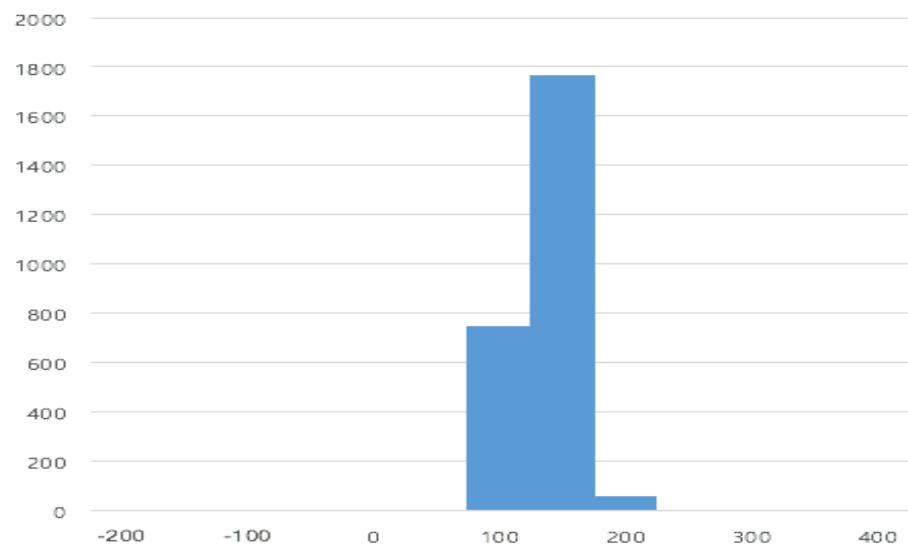


→ Epoch 50 일 때, 정상 데이터 Output

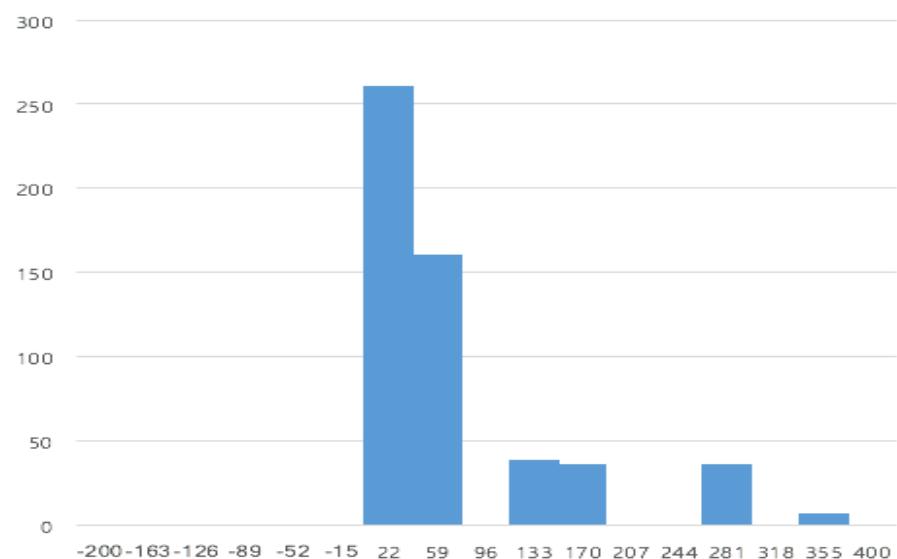


→ Epoch 50 일 때, 비정상 데이터 Output

Epoch 가 50 일 때는 502-2 Case 보다 정상 비정상을 잘 구분함.



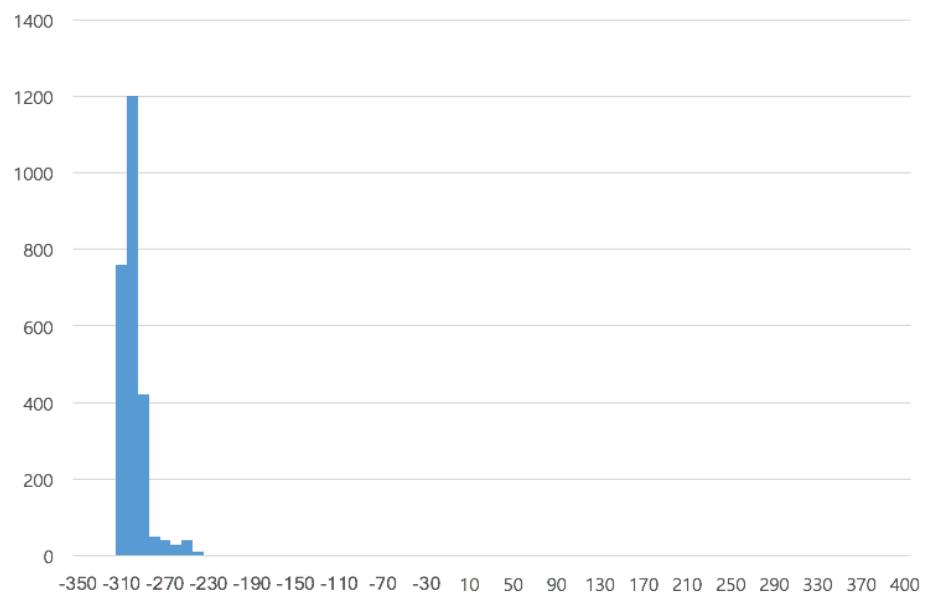
→ Epoch 1200 일 때, 정상 데이터 Output



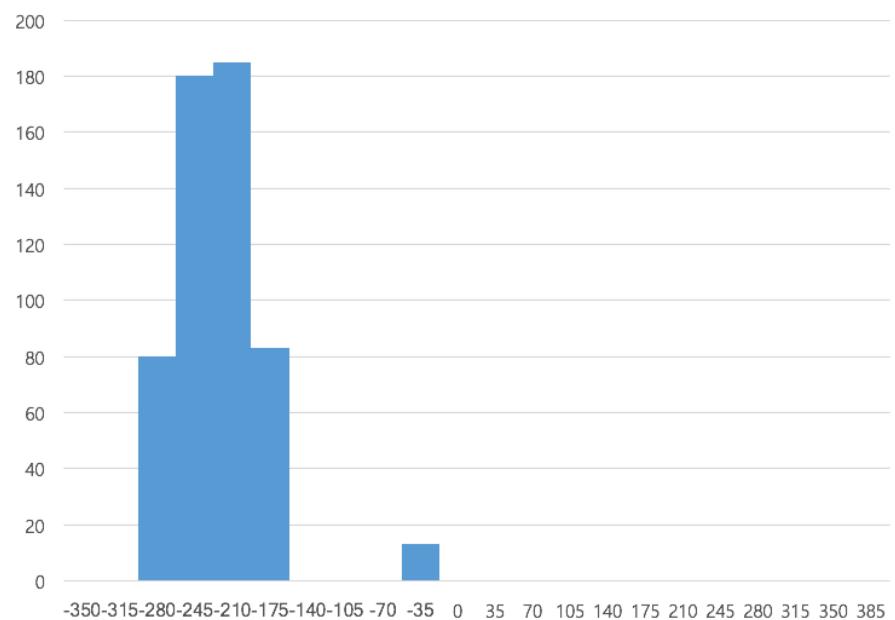
→ Epoch 1200 일 때, 비정상 데이터 Output

Epoch 1200 일 때는 100 을 기준으로 정상 비정상 데이터를 구분함.

마지막으로 정확도가 85%인 GPU model 이다.

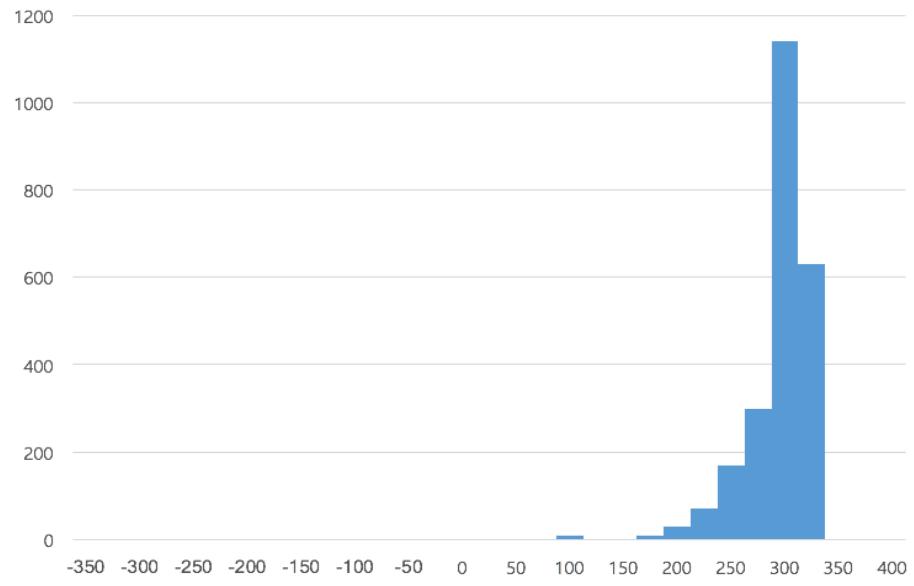


→ Epoch 50 일 때, 정상 데이터 Output

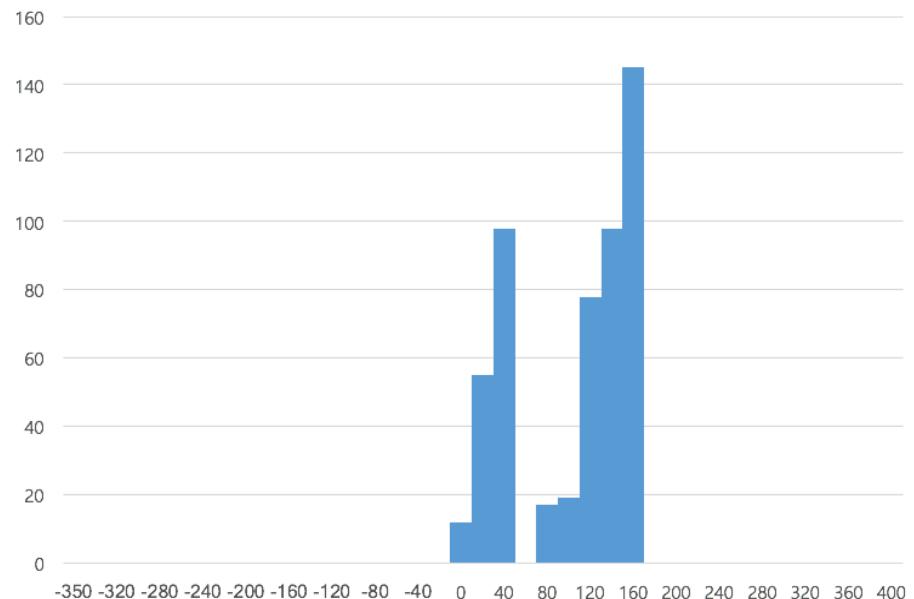


→ Epoch 50 일 때, 비정상 데이터 Output

Epoch 가 50 일 때는 -270 을 기준으로 정상과 비정상을 구분함.



→ Epoch 1100 일 때, 정상 데이터 Output



→ Epoch 1100 일 때, 비정상 데이터 Output

Epoch 가 1100 일 때는, 160 을 기준으로 정상과 비정상을 구분함.

4.3 개발환경

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

<Table 3. 개발환경>

5. 프로젝트 수행 계획

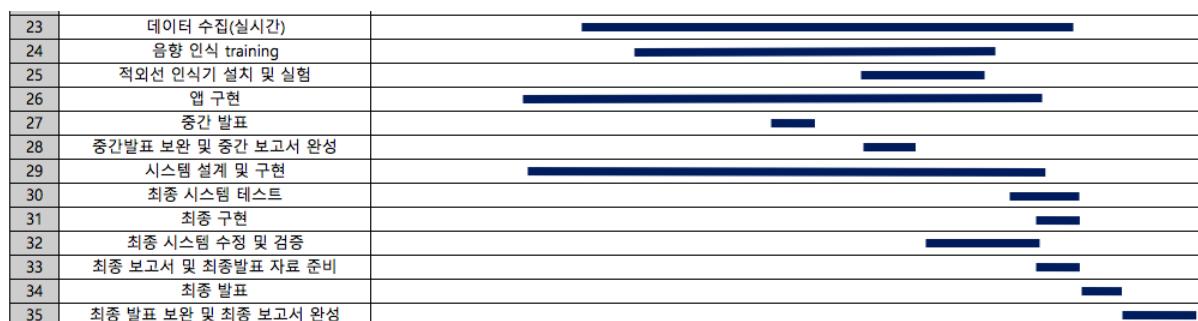
5.1 업무 분담

이승진(PM)	강인선	오익준	이정하
- 프로젝트 일정관리	- 음향 데이터 분석	- 회의록 작성	- PPT 제작
- GAN 기법 스터디	- 실시간 데이터	- EP 시스템 구축	- 문서 관리
- 데이터 전처리	- neural network 구축	- App 구축	- 적외선 센서(Arduino)
- DB 구축	& data training		- EP 시스템 구축
			- 데이터 정리

<Table 4. 업무 분담표>

5.2 향후 일정

7.2.1 간트 차트



<Figure 14. 간트 차트>

7.2.2 회의 일정

회의 내용	회의 시간	회의 장소
팀 회의	매주 월, 수, 금 8PM~	공학관 502 호
지도 교수님 면담	매주 화요일 12:30PM~1:20PM	공학관 530 호

<Table 5. 회의 일정>

6. 참고자료

- 라즈베리파이 쿡북, 사이먼 몽크, 박경욱, 백운혁, 유시형 한빛 미디어 (2015)
 - 모두의 라즈베리파이 with Python, 이시이 모루나, 에사키 노리히데, 길벗 (2016)
 - 음악신호 머신러닝 초심자를 위한 가이드, <http://keunwoochoi.blogspot.kr>
 - 파이썬 라이브러리를 활용한 머신러닝, 안드레아스 뮐러, 세라 가이도 한빛 아카데미 (2017)
 - Unsupervised classification of respiratory sound signal into snore/no-snore classes2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Azarbarzin, Al
 - Unsupervised classification of heart sound recordings2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference Tsai, Wei-Ho
 - 적외선 거리 측정 센서를 이용한 보행자 수 측정 2008.6, 819-820
 - 적외선 센서 기반의 사람/차량 탐지 적용 알고리즘An Adaptive Person/Vehicle Detection Algorithm for PIR Sensor김영만 식별저자, 박장호, 김이형, 박홍재 정보과학회논문지
 - Introduction to Speech Processing | Ricardo Gutierrez-Osuna | CSE@TAMU
 - Machine Learning Engineer and Chief Unicorn Scientist. Global Shaper at World Economic Forum. English, French, German, Arabic, and Japanese speaker.
 - Non-sequential automatic classification of anuran sounds for the estimation of climate-change indicatorsAuthor AmaliaLuquea,Javier Romero-Lemos , Alejandro Carrascob,Julio Barbanchob
 - Stanford University CS231n: Convolutional Neural Networks for Visual Recognition spring (2017)
 - <https://kocoafab.cc/make/view/173> , 동전에 반응하는 저금통(2014)
 - <https://blog.naver.com/mchoi0602/220918629042>, 아두이노 강좌(2017)
 - <https://arxiv.org/abs/1406.2661>
 - <https://arxiv.org/abs/1701.00160>
 - https://www.youtube.com/watch?v=odpjk7_tGY0
 - Arjovsky, Martin, Chintala, Soumith, and Bottou, Le' on. Wasserstein GAN. In ICML, 2017.
 - Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville. Improved Training of Wasserstein GANs
 - Chris Donahue, Julian McAuley , Miller Puckette. Synthesizing Audio with Generative Adversarial Networks. 2018
 - 코딩셰프의 3분 딥러닝 케라스맛, 김성진 지음 (2018)
 - https://www.youtube.com/channel/UChflhu32f5EUHlY7_SetNWw 안드로이드 스튜디오 강좌
 - <https://www.youtube.com/channel/UCSci4bsr-RaGdYSC2QAHWug/playlists>
- 안드로이드 앱 개발 강의

7. 별첨

7.1 포스 벽진



- 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 외주진행 제품 (10days)
- 월 평균 판매량 : 700EA
- 일 평균 생산대수 : 30EA
- 일 평균 생산 금액 : 3,000,000(₩)
(RACK 제조비용 평균 : ₩100,000 / EA)

<Figure 15. 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 6. 기업 개요>

생
산
시
설



Punching Machine



Bending Machine



<Figure 17. 주요 제품 RACK>

<Figure 16. 기계 종류>

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

<Table 7. 공정 순서도>

7.2 금형 종류



<Figure 18. 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 8. 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

7.3 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 9. 철판 종류 표>

7.4 Generate Adversarial Network (GAN) Objective Function

$$\min_G \max_D V(D, G) = E_{x \sim p_{\text{data}}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

Sample x from real data distribution Sample latent code z from Gaussian distribution
 ↓ ↓
 Maximum when $D(x) = 1$ Maximum when $D(G(z)) = 0$
 ↗ ↗
 D should maximize $V(D, G)$

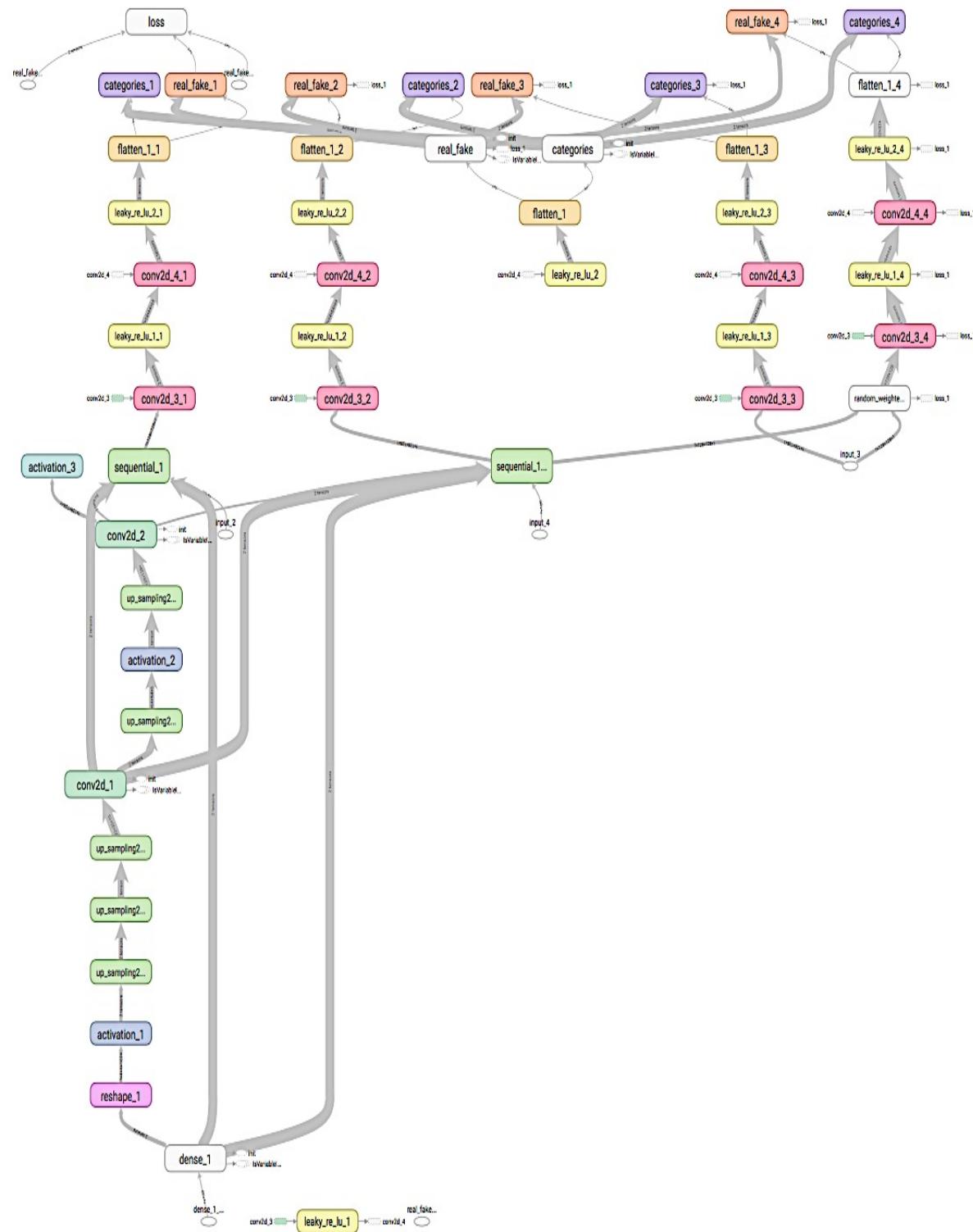
→ Discriminator should maximize

$$\min_G \max_D V(D, G) = \cancel{E_{x \sim p_{\text{data}}(x)}[\log D(x)]} + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

G is independent of this part ↓
 ↗ ↗
 G should minimize $V(D, G)$ Minimum when $D(G(z)) = 1$

→ Generator should minimize

7.5 Hidden Layer 구조도



<Figure 19. Hidden layer 구조도>