

데이터분석 캡스톤 디자인 최종발표

종이시험지 자동 채점 프로그램

발표자	박나연
-----	-----

학번	2	0	1	9	1	0	2	0	9	4
----	---	---	---	---	---	---	---	---	---	---

1. 다음 보기 중 프로젝트를 진행한 달을 모두 고르시오.

① 2021년 8월

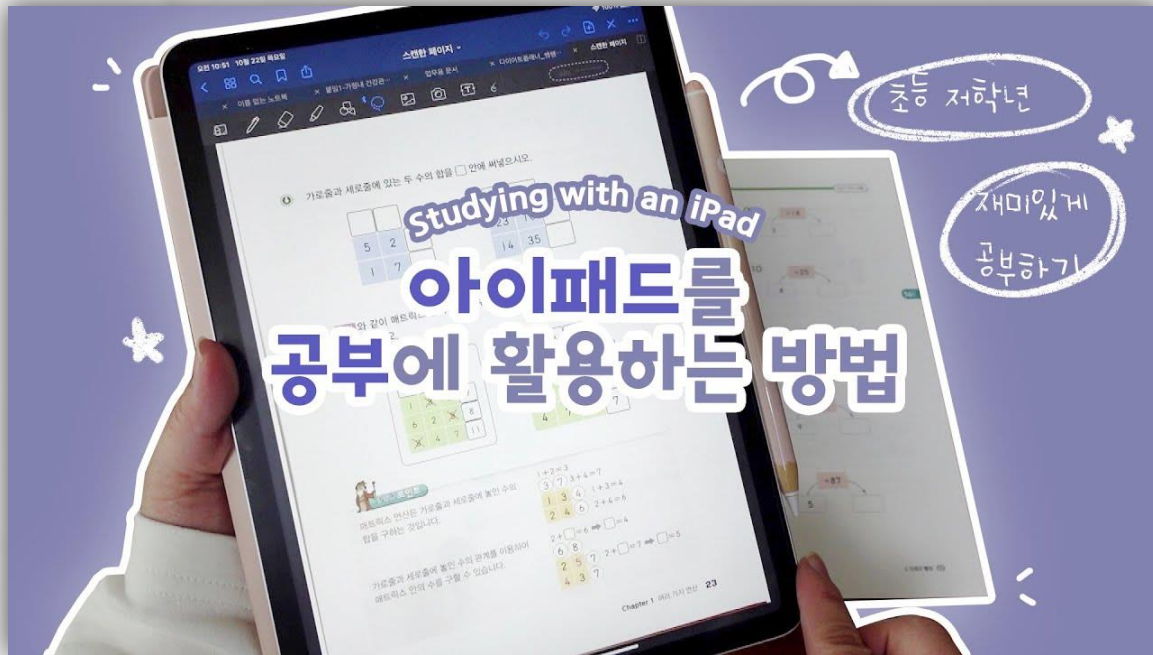
② 2021년 9월

③ 2021년 10월

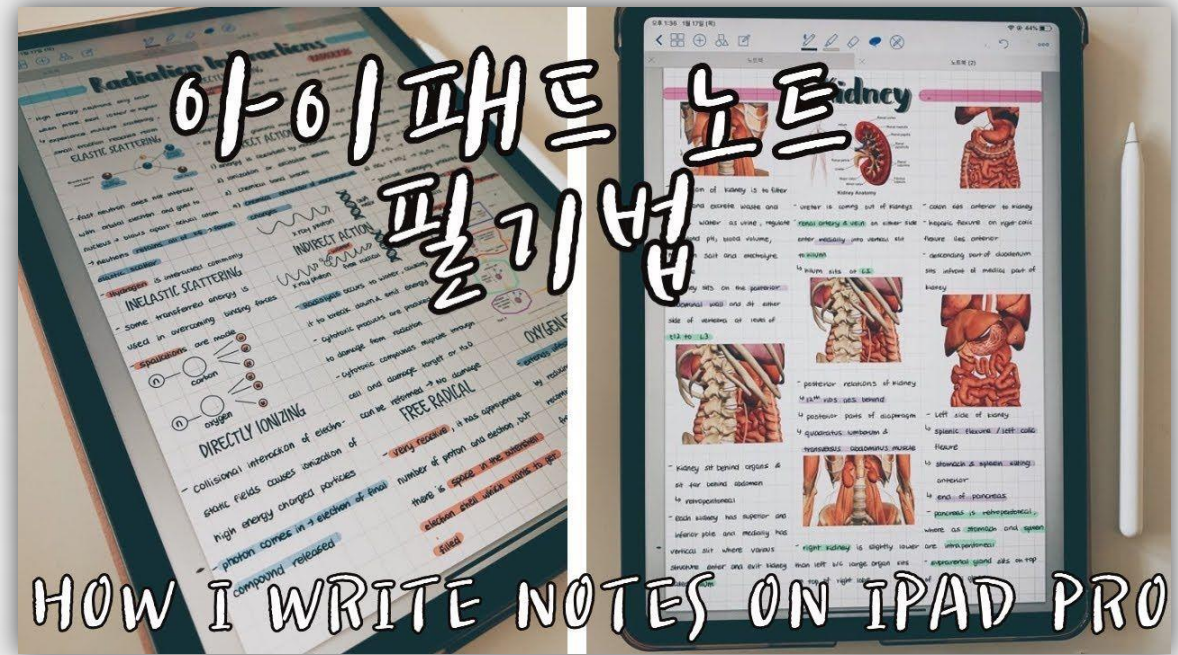
④ 2021년 11월

⑤ 2021년 12월

⑥ 2022년 1월



태블릿PC로 공부하는 학생들 증가



태블릿PC를 활용한 모바일 앱 서비스 증가

기쁘기쁘

아이패드로 수능 기출 뽐시기



문제집도

애플 펜슬로 문제 풀고 수능 - 집도

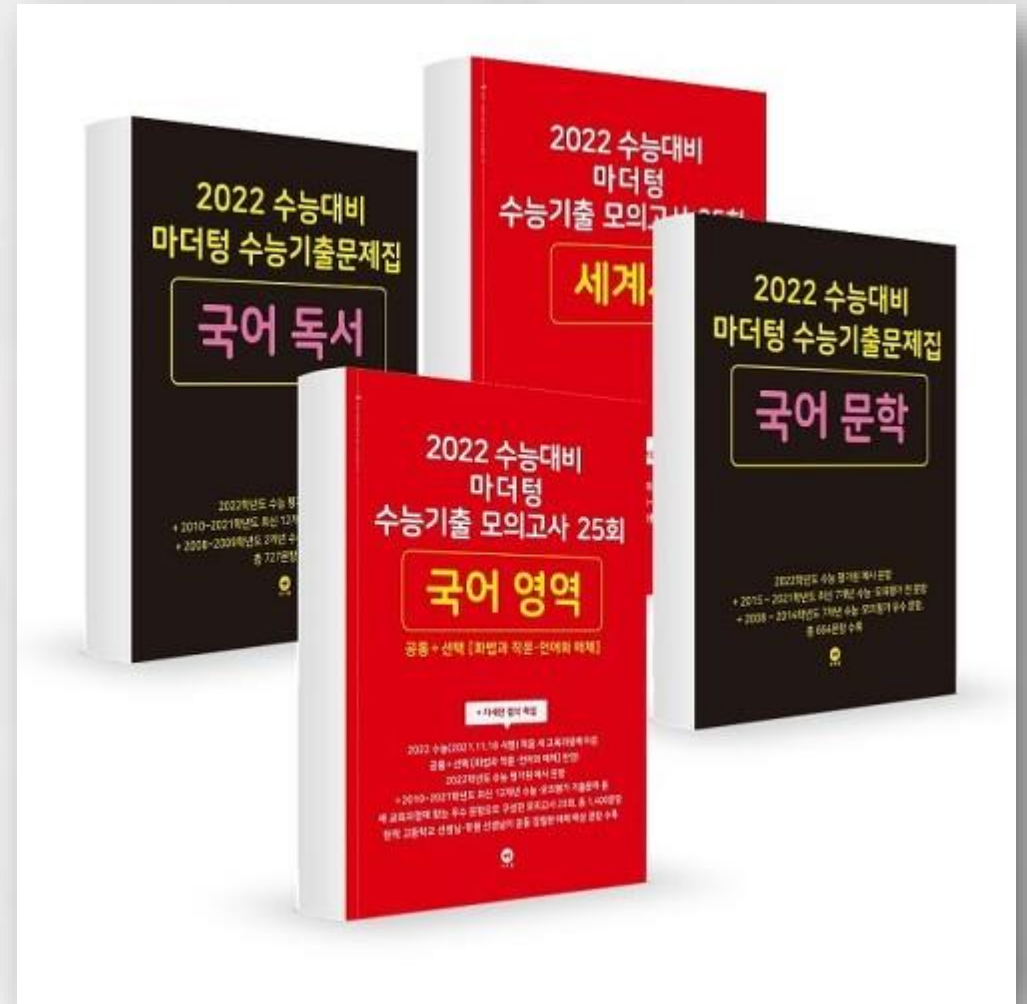


자동채점

채점은 내가 맡는다, 넌 풀기만 해



그러나 여전히 ...



그래서

종이 시험지를 위한 ai서비스를 기획해
편리한 학습의 범위를 확장,

종이시험지 자동 채점 프로그램

최종 목표

많은 양의 시험지를 눈으로 보고 채점하는 것 대신, 문제를 자동으로 인식해 채점해주자!

1. 시험지 촬영

기우개

2021학년도 대학수학능력시험 문제지 1

제 2 교시 수학 영역(가형) 출수형

5지선다형

1. $\sqrt{6} \times 3^{\frac{1}{2}}$ 의 값은? [2점]

Ⓐ 1 Ⓑ $3^{\frac{1}{2}}$ Ⓒ 3 Ⓓ $3^{\frac{3}{2}}$ **Ⓔ 6**

2. $\lim_{n \rightarrow \infty} \frac{1}{\sqrt{4n^2+2n+1}-2n}$ 의 값은? [2점]

Ⓐ 1 Ⓑ 2 Ⓒ 3 **Ⓓ 4** Ⓔ 5

3. $\frac{\pi}{2} < \theta < \pi$ 인 θ 에 대하여 $\sin \theta = -\frac{\sqrt{3}}{7}$ 일 때, $\tan \theta$ 의 값은? [2점]

Ⓐ $-\frac{\sqrt{3}}{2}$ **Ⓑ $-\frac{\sqrt{3}}{4}$** Ⓒ 0 Ⓓ $\frac{\sqrt{3}}{4}$ Ⓔ $\frac{\sqrt{3}}{2}$

4. 두 사건 A, B에 대하여 $P(B|A) = \frac{1}{4}$, $P(A|B) = \frac{1}{3}$, $P(A) + P(B) = \frac{7}{10}$ 일 때, $P(A \cap B)$ 의 값은? [2점]

Ⓐ $\frac{1}{7}$ Ⓑ $\frac{1}{8}$ Ⓒ $\frac{1}{9}$ Ⓓ $\frac{1}{10}$ Ⓔ $\frac{1}{11}$



2. 배경과 판권

2021학년도 대학수학능력시험 문제지 1

제 2 교시 수학 영역(가형) 출수형

5지선다형

1. $\sqrt{6} \times 3^{\frac{1}{2}}$ 의 값은? [2점]

Ⓐ 1 Ⓑ $3^{\frac{1}{2}}$ Ⓒ 3 Ⓓ $3^{\frac{3}{2}}$ **Ⓔ 6**

2. $\lim_{n \rightarrow \infty} \frac{1}{\sqrt{4n^2+2n+1}-2n}$ 의 값은? [2점]

Ⓐ 1 Ⓑ 2 Ⓒ 3 **Ⓓ 4** Ⓔ 5

3. $\frac{\pi}{2} < \theta < \pi$ 인 θ 에 대하여 $\sin \theta = -\frac{\sqrt{3}}{7}$ 일 때, $\tan \theta$ 의 값은? [2점]

Ⓐ $-\frac{\sqrt{3}}{2}$ **Ⓑ $-\frac{\sqrt{3}}{4}$** Ⓒ 0 Ⓓ $\frac{\sqrt{3}}{4}$ Ⓔ $\frac{\sqrt{3}}{2}$

4. 두 사건 A, B에 대하여 $P(B|A) = \frac{1}{4}$, $P(A|B) = \frac{1}{3}$, $P(A) + P(B) = \frac{7}{10}$ 일 때, $P(A \cap B)$ 의 값은? [2점]

Ⓐ $\frac{1}{7}$ Ⓑ $\frac{1}{8}$ Ⓒ $\frac{1}{9}$ Ⓓ $\frac{1}{10}$ Ⓔ $\frac{1}{11}$

이 문항지에 관한 정답은 한국교육과정평가원에 있습니다.

3. 문제 인식

제 2 교시

수학 영역(가) 학

출수형

class 1

1. $\sqrt{9} \times 3^{\frac{1}{2}}$ 의 값은? [2점]

- Ⓐ 1 Ⓑ $3^{\frac{1}{2}}$ Ⓒ 3 Ⓓ $3^{\frac{3}{2}}$ Ⓔ $\sqrt{9}$

class 1

3. $\frac{\pi}{2} < \theta < \pi$ 인 θ 에 대하여 $\sin \theta = -\frac{\sqrt{21}}{7}$ 인 때, $\tan \theta$ 의 값은? [2점]

- Ⓐ $-\frac{\sqrt{3}}{2}$ Ⓑ $-\frac{\sqrt{3}}{4}$ Ⓒ 0 Ⓓ $\frac{\sqrt{3}}{4}$ Ⓔ $\frac{\sqrt{3}}{2}$

class 1

4. 두 사건 A, B에 대하여

$$P(B|A) = \frac{1}{4}, P(A|B) = \frac{1}{3}, P(A) + P(B) = \frac{7}{10}$$

인 때, $P(A \cap B)$ 의 값은? [3점]

- Ⓐ $\frac{1}{7}$ Ⓑ $\frac{1}{8}$ Ⓒ $\frac{1}{9}$ Ⓓ $\frac{1}{10}$ Ⓔ $\frac{1}{11}$

class 1

2. $\lim_{n \rightarrow \infty} \frac{1}{\sqrt{4n^2 + 2n + 1} - 2n}$ 의 값은? [2점]

- Ⓐ 1 Ⓑ 2 Ⓒ 3 Ⓓ 4 Ⓔ 6

4. 선택항목 인식

1. $\sqrt{9} \times 3^{\frac{1}{2}}$ 의 값은? [2점]

class1

- Ⓐ 1

class1

- Ⓑ $3^{\frac{1}{2}}$

class1

- Ⓒ 3

class1

- Ⓓ $3^{\frac{3}{2}}$

class2

- Ⓔ $\sqrt{9}$

5. 선택한 번호 + 틀린문제 출력

1번 : 5

2번 : 4

3번 : 2

4번 : 1

5번 : 2

6번 : 3

7번 : 4

8번 : 3

.

.

.

7번의 정답 : 2 선택한 답 : 4

13번의 정답 : 4 선택한 답 : 1

6. 종료

진행 내용

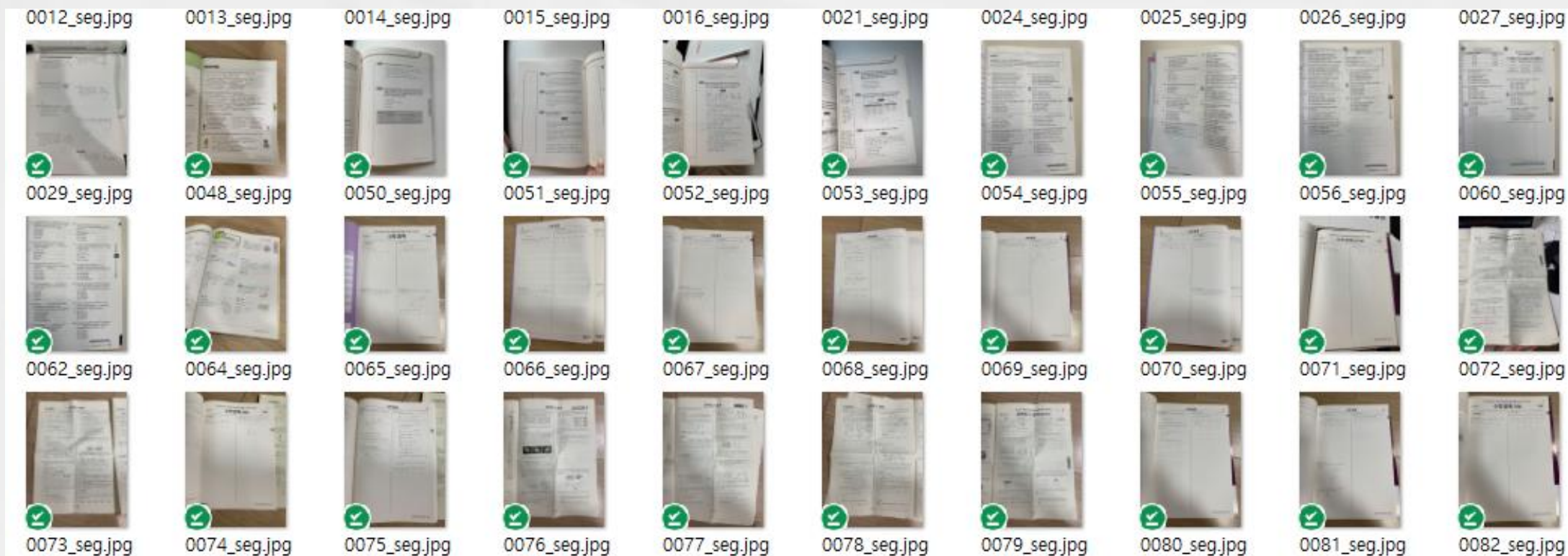
데이터 준비	모델 학습	모델 테스트
<ul style="list-style-type: none">① 데이터 수집 ② 데이터 라벨링 ③ TFRecord 생성	<ul style="list-style-type: none">① 배경분리 모델 training ② 문제 감지 모델 training ③ 항목 감지 모델 training	<ul style="list-style-type: none">① 배경 분리 모델 적용 후 Perspective Transform ② 문제 감지 모델 적용 후 문제 별로 자르기 ③ 항목 감지 모델 적용 후 선택한 항목의 번호 파악

데이터 준비

데이터 수집과 라벨링

데이터 수집

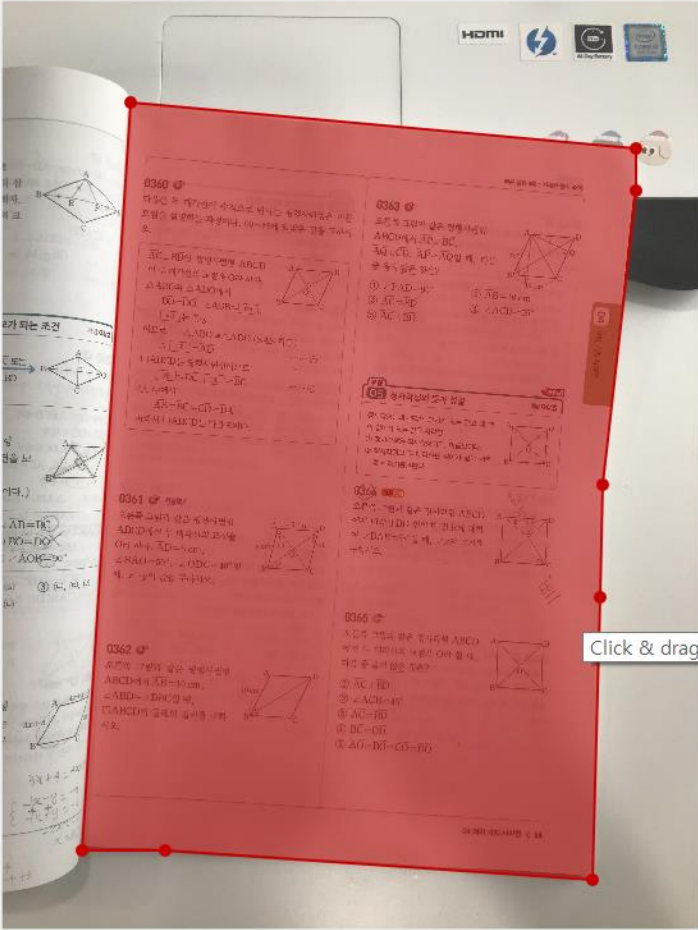
배경분리를 위한 촬영 이미지



Labeling

labelme - C:/Users/mycom/Desktop/자료/데분캡디/data/0003.jpg

File Edit View Help



Flags

Label List

Polygon Labels

File List

```
"version": "4.5.12",  
"flags": {},  
"shapes": [  
  {  
    "label": "paper",  
    "points": [  
      [
```



0001_seg.json

```
    "label": "paper",
```

```
    "points": [  
      [
```

```
        188.304347826087,  
        329.9130434782609
```

```
      ],
```

```
      [
```

```
        279.608695652174,  
        3856.0
```

```
      ],
```

```
      [
```

```
        2931.782608695652,  
        3725.565217391304
```

```
      ],
```

```
      [
```

```
        2783.95652173913,  
        334.2608695652174
```

```
      ]
```

```
    ],
```

```
    "group_id": null,
```

```
    "shape_type": "polygon",
```

```
    "flags": {}
```

```
  ]
```

Click & drag to move shape 'paper'

데이터 수집

수능 시험지 데이터



다양한 영역(중학시험, 검정고시, 경찰대학, 공무원) 시험지 395장 수집
그 중 수능시험지 100장

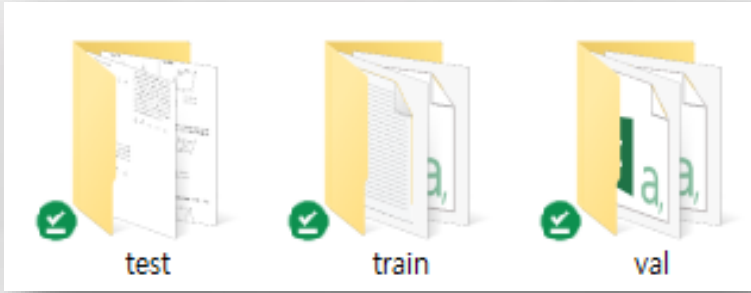
데이터 수집

수능 시험지 데이터

Augmentation

밝기조절, 노이즈 추가, 기울이기



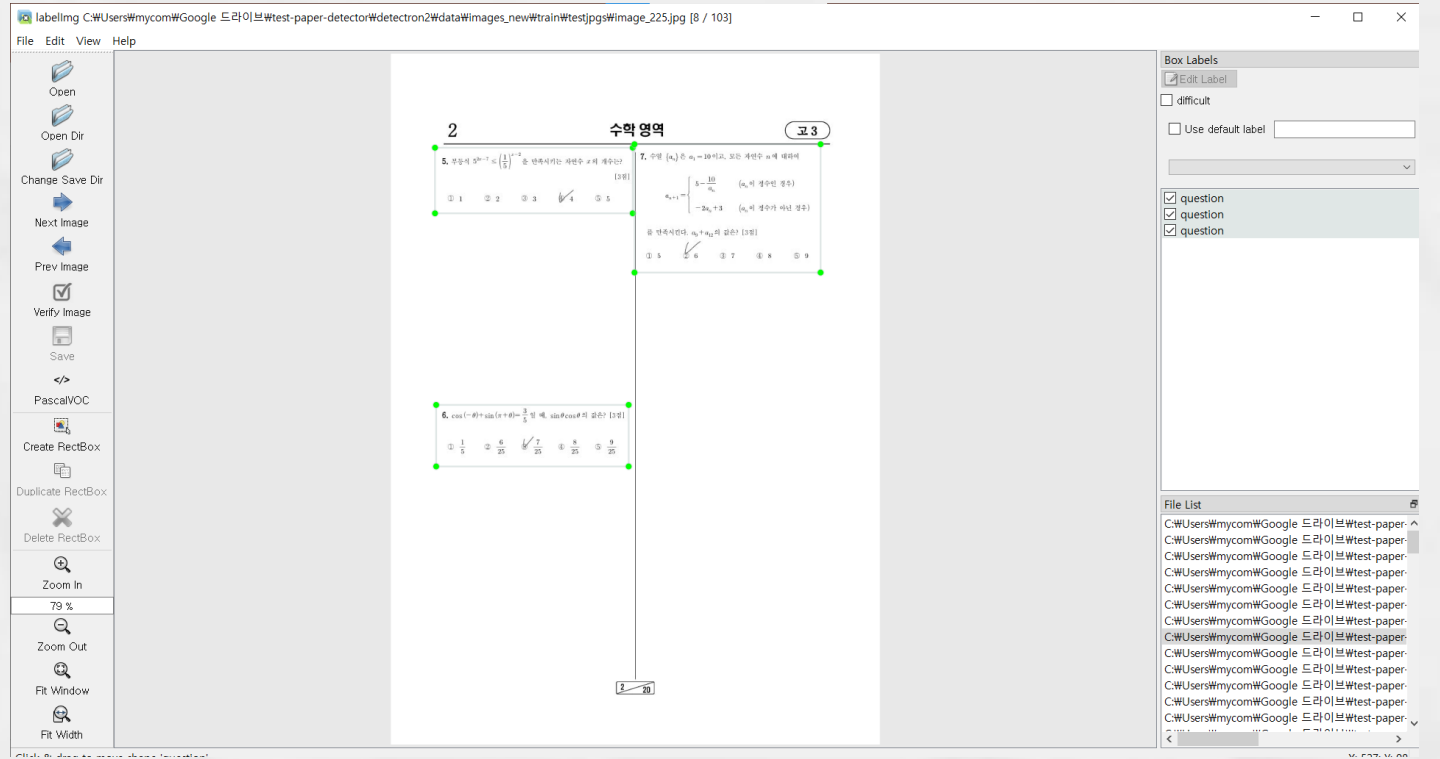


Train 약 400장

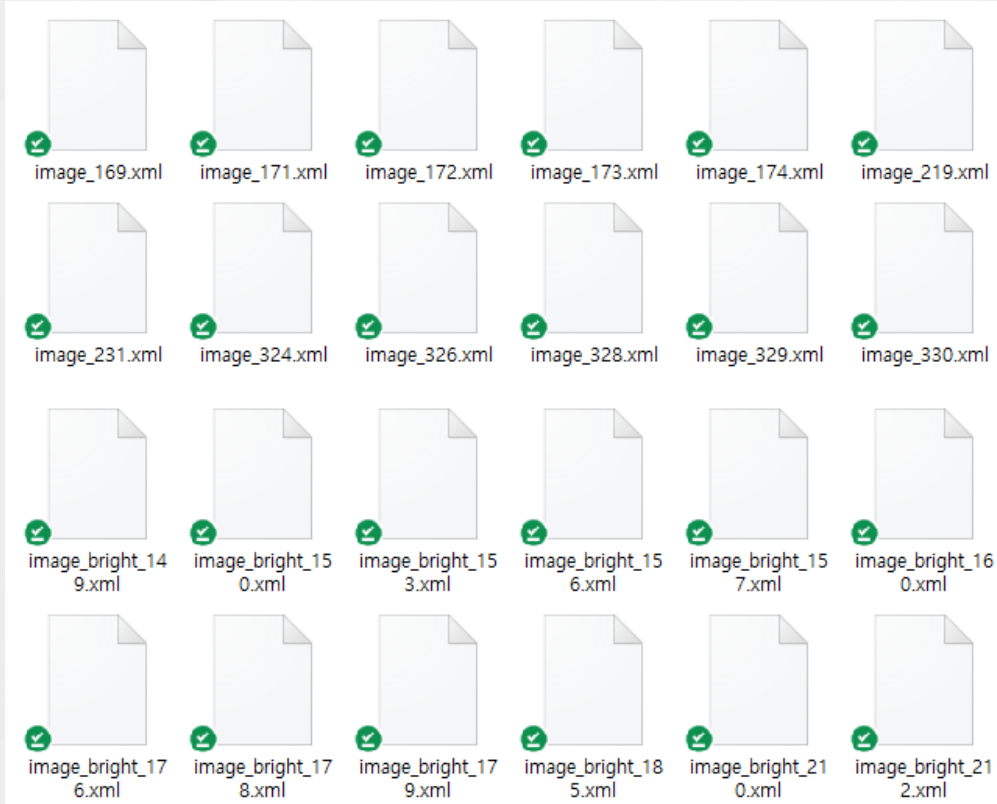
Val 약 200장

Test 약 200장

라벨링 툴 LabelImg로 문제 라벨링 진행



각 이미지에 대한 라벨링 정보를 담은 xml파일



```
1 <annotation>
2 <folder>testjpgs</folder>
3 <filename>image_169.jpg</filename>
4 <path>C:\Users\mycom\Google 드라이브\test-paper-detector\detectron2\data\images_new\train\testjpgs\image_169.jpg</path>
5 <source>
6 <database>Unknown</database>
7 </source>
8 <size>
9 <width>595</width>
10 <height>841</height>
11 <depth>3</depth>
12 </size>
13 <segmented>0</segmented>
14 <object>
15 <name>question</name>
16 <pose>Unspecified</pose>
17 <truncated>0</truncated>
18 <difficult>0</difficult>
19 <bndbox>
20 <xmin>25</xmin>
21 <ymin>212</ymin>
22 <xmax>270</xmax>
23 <ymax>279</ymax>
24 </bndbox>
25 </object>
26 <object>
27 <name>question</name>
28 <pose>Unspecified</pose>
29 <truncated>0</truncated>
30 <difficult>0</difficult>
31 <bndbox>
32 <xmin>22</xmin>
33 <ymin>444</ymin>
34 <xmax>279</xmax>
35 <ymax>533</ymax>
36 </bndbox>
37 </object>
38 <object>
39 <name>question</name>
40 <pose>Unspecified</pose>
41 <truncated>0</truncated>
42 <difficult>0</difficult>
43 <bndbox>
44 <xmin>297</xmin>
45 <ymin>159</ymin>
46 <xmax>555</xmax>
47 <ymax>278</ymax>
48 </bndbox>
49 </object>
50 <object>
51 <name>question</name>
52 <pose>Unspecified</pose>
53 <truncated>0</truncated>
54 <difficult>0</difficult>
55 <bndbox>
56 <xmin>298</xmin>
```

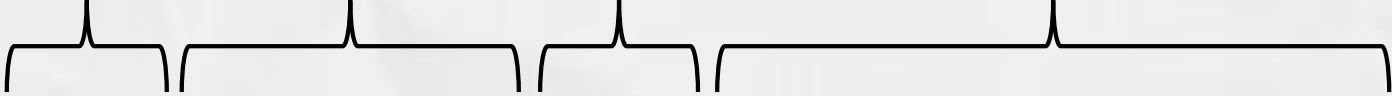
xml파일 속 정보를 csv 파일로 병합

이미지 파일 name

이미지 size

Class name

Bounding box 좌표

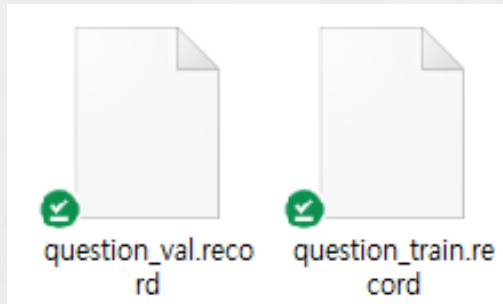


	filename	width	height	class	xmin	ymin	xmax	ymax
1	filename	width	height	class	xmin	ymin	xmax	ymax
2	image_169	595	841	question	25	212	270	279
3	image_169	595	841	question	22	444	279	533
4	image_169	595	841	question	297	159	555	278
5	image_169	595	841	question	298	439	555	607
6	image_171	595	841	question	23	74	283	208
7	image_171	595	841	question	22	404	294	484
8	image_171	595	841	question	297	69	569	181
9	image_171	595	841	question	298	397	568	626
10	image_172	595	841	question	23	70	293	228
11	image_172	595	841	question	21	390	298	543
12	image_172	595	841	question	295	70	568	191
13	image_173	595	841	question	22	73	296	209
14	image_173	595	841	question	25	422	294	534
15	image_173	595	841	question	297	68	546	190

Train set / Validation set 의 TFRecord 파일 생성

Tensorflow Object Detection API 에서 제공하는 `generate_tfrecord.py` 사용

```
! python research/generate_tfrecord.py --csv_input=data/images_new/val/val_labels.csv --output_path=research/object_detection/data/question_val.record --i
Successfully created the TFRecords: /content/drive/My Drive/test-paper-detector/detectron2/research/object_detection/data/question_val.record
```



Label map 파일 생성

```
1 item {
2     id: 1
3     name: 'question'
4 }
```

문제 감지를 위한 라벨맵

```
1 item {
2     id: 1
3     name: 'not-choice'
4 }
5 item {
6     id: 2
7     name: 'choice'
8 }
```

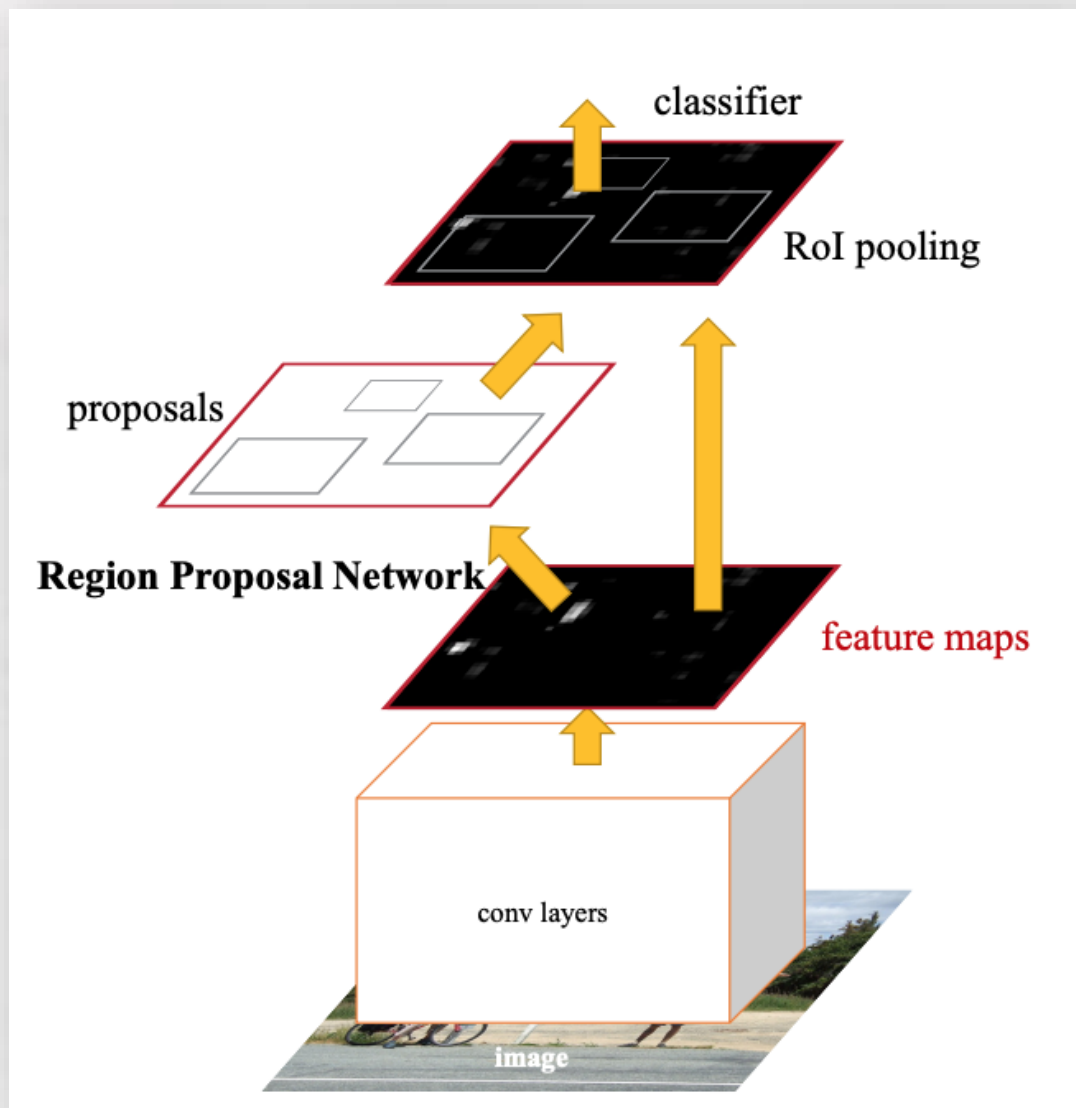
항목 감지를 위한 라벨맵

모델 학습

Tensorflow Object Detection

사용되는 모델

Faster R-CNN

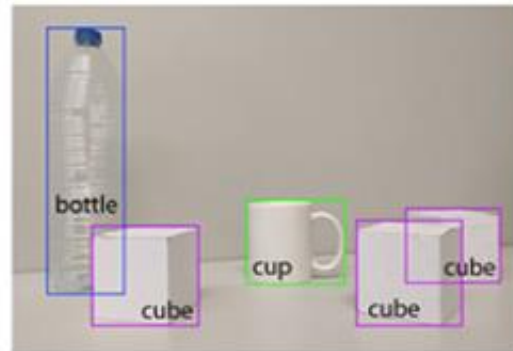


사용되는 모델

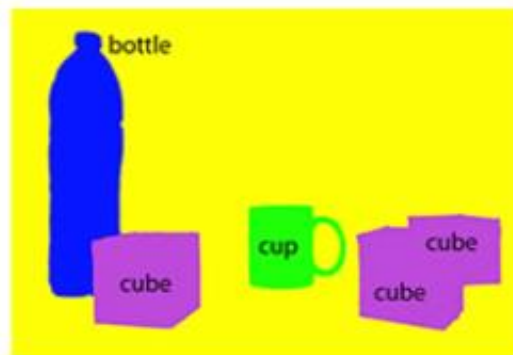
Mask R-CNN



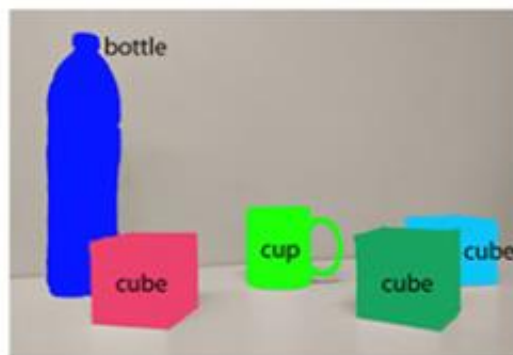
(a) Image classification



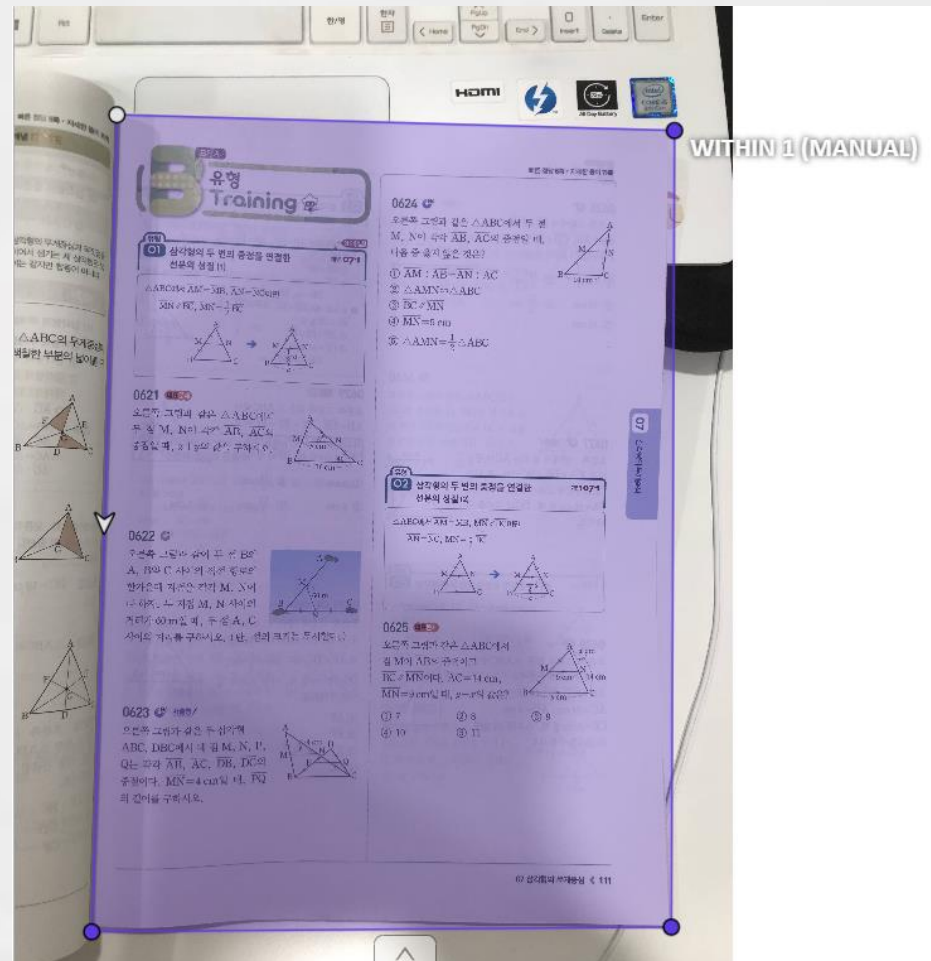
(b) Object localization



(c) Semantic segmentation

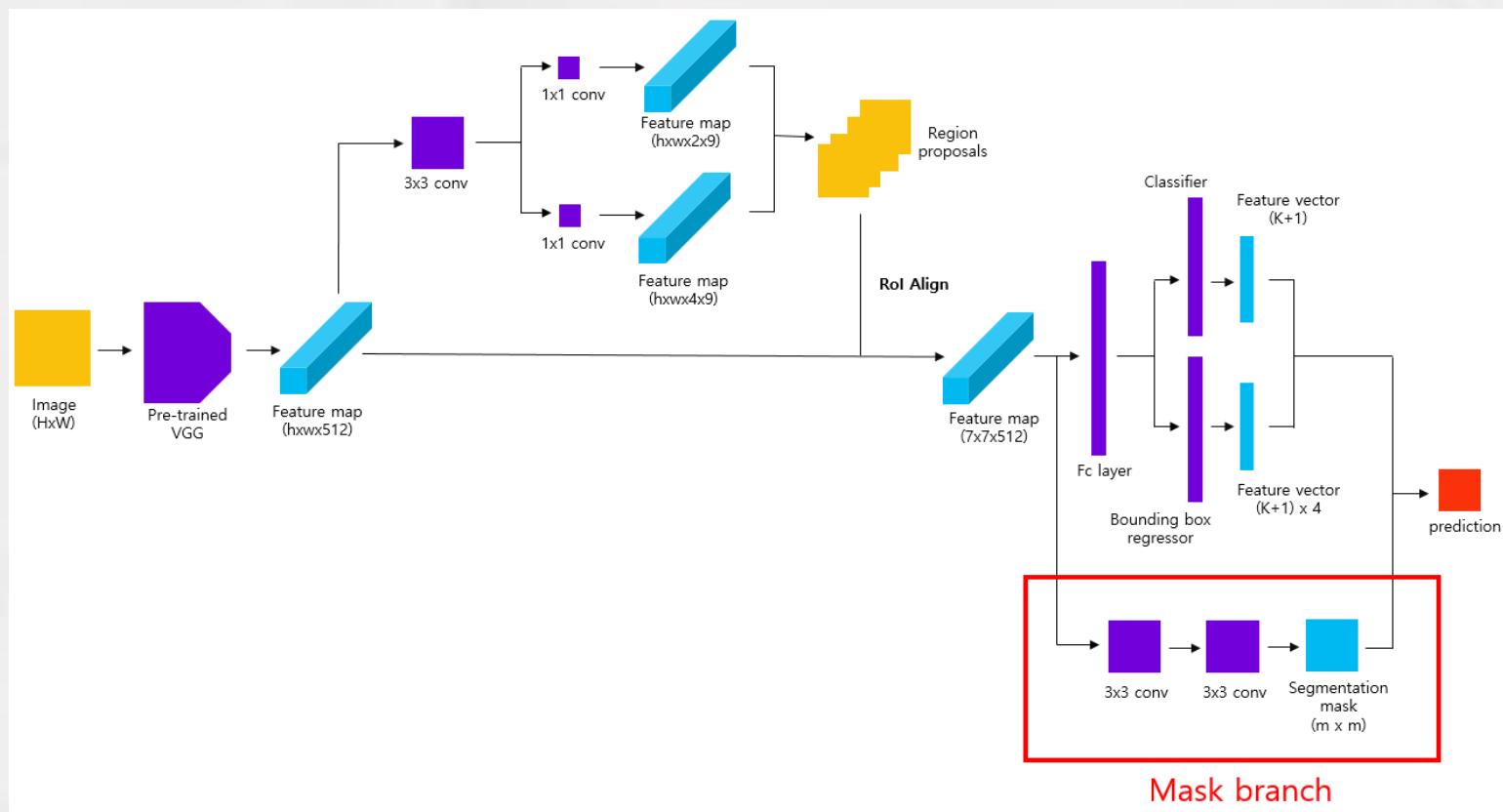


(d) Instance segmentation



사용되는 모델

Mask R-CNN



Tensorflow Object Detection API

미리 학습된 모델

모델 이름	속도	COCO mAP
ssd_mobilenet_v1_coco 다운로드	빠름	21
ssd_inception_v2_coco 다운로드	빠름	24
rfcn_resnet101_coco 다운로드	보통	30
faster_rcnn_resnet101_coco 다운로드	보통	32
faster_rcnn_inception_resnet_v2_atrous_coco 다운로드	느림	37

Git Clone을 통해 설치

tensorflow / models Public

Watch 2.9k


Fork 45.3k

Star 72.3k

Code Issues 1.1k Pull requests 171 Actions Projects 5 Wiki Security Insights

master models / research / object_detection /

Go to file Add file ...

	tensorflow-gardener and TF Object Detection Team fixing documentation on how to use object detection model on Android ...	4e95c87 3 days ago	History
..			
anchor_generators	Release MobileDet code and model, and require tf_slim installation fo...		2 years ago
box_coders	Release MobileDet code and model, and require tf_slim installation fo...		2 years ago
builders	Add SpaghettiNet Feature Extractor		2 months ago
colab_tutorials	Generalize the xml parsing to facilitate custom datasets.		29 days ago
configs	Add SpaghettiNet Feature Extractor		2 months ago
core	Implement bounding box tightness prior and CC loss warmup.		7 days ago
data	Release MobileDet code and model, and require tf_slim installation fo...		2 years ago
data_decoders	Updated the tf_example_decoder to support the case where the keypoint...		2 months ago

디렉토리 구성

데이터 파일과 object detection 폴더를 포함해 아래와 같이 디렉토리를 구성한다

```
|— research
|   |— object_detection
|   |   |— samples
|   |   |   |— configs
|   |   |   |   |— faster_rcnn_inception_resnet_v2_atrous_coco.config
|   |   |   |   └─ checkpoint
|   |   |— data
|   |   |   |— question.record
|   |   |   |— choice.record
|   |   |   |— question_labelmap.pbtxt
|   |   |   └─ choice_labelmap.pbtxt
|   |— faster_rcnn_inception_resnet_v2_atrous_coco_2017_11_08
|   |   |— frozen_inference_graph.pb
|   |   └─ model.ckpt
|   |— train.py
|   |— export_inference_graph.py
|   └─ generate_tfrecord.py
```

디렉토리 구성

데이터 파일과 object detection 폴더를 포함해 아래와 같이 디렉토리를 구성한다

```
├─ research
│  └─ object_detection
│     └─ samples
│        └─ configs
│           └─ faster_rcnn_inception_resnet_v2_atrous_coco.config
│           └─ checkpoint
│     └─ data
│        └─ question.record
│        └─ choice.record
│        └─ question_labelmap.pbtxt
│        └─ choice_labelmap.pbtxt
├─ faster_rcnn_inception_resnet_v2_atrous_coco_2017_11_08
│  └─ frozen_inference_graph.pb
│  └─ model.ckpt
├─ train.py
├─ export_inference_graph.py
└─ generate_tfrecord.py
```

Config 파일 수정

앞에서 생성한 데이터 파일 수정
감지할 클래스 개수 수정
미리 학습된 모델의 체크포인트 추가

```
1 model {
2   faster_rcnn {
3     num_classes: 1
4     image_resizer {
5       keep_aspect_ratio_resizer {
6         min_dimension: 600
7         max_dimension: 1024
8       }

```

```
100   gradient_clipping_by_norm: 10.0
101   fine_tune_checkpoint: "faster_rcnn_inception_resnet_v2_atrous_coco_2017_11_08/model.ckpt"
102   from_detection_checkpoint: true

```

```
114 train_input_reader: {
115   tf_record_input_reader {
116     input_path: "object_detection/data/question_train.record"
117   }
118   label_map_path: "object_detection/data/question_labelmap.pbtxt"
119 }

```

```
128 eval_input_reader: {
129   tf_record_input_reader {
130     input_path: "object_detection/data/question_val.record"
131   }
132   label_map_path: "object_detection/data/question_labelmap.pbtxt"

```


문제 감지 모델 학습

수정한 config 파일과 제공되는 **train.py**로 학습 진행

```
!python train.py --train_dir=object_detection/samples/configs/ --pipeline_config_path=object_detection/samples/configs/faster_rcnn_inception_resnet_v2_atrous_coco.config
```

```
INFO:tensorflow:global step 1170: loss = 0.1704 (3.467 sec/step)
INFO:tensorflow:global step 1171: loss = 0.1443 (3.468 sec/step)
INFO:tensorflow:global step 1172: loss = 0.1546 (3.502 sec/step)
INFO:tensorflow:global step 1173: loss = 0.1246 (3.479 sec/step)
INFO:tensorflow:global step 1174: loss = 0.1258 (3.479 sec/step)
INFO:tensorflow:global step 1175: loss = 0.1903 (3.489 sec/step)
INFO:tensorflow:global step 1176: loss = 0.0835 (3.497 sec/step)
INFO:tensorflow:global step 1177: loss = 0.0943 (3.501 sec/step)
INFO:tensorflow:global step 1178: loss = 0.0917 (3.509 sec/step)
INFO:tensorflow:global step 1179: loss = 0.1669 (3.503 sec/step)
INFO:tensorflow:global step 1180: loss = 0.0699 (3.493 sec/step)
INFO:tensorflow:global step 1181: loss = 0.1535 (3.481 sec/step)
INFO:tensorflow:global step 1182: loss = 0.1174 (3.506 sec/step)
INFO:tensorflow:global step 1183: loss = 0.3306 (3.512 sec/step)
INFO:tensorflow:global step 1184: loss = 0.1134 (3.491 sec/step)
INFO:tensorflow:global step 1185: loss = 0.1758 (3.535 sec/step)
INFO:tensorflow:global step 1186: loss = 0.1583 (3.547 sec/step)
INFO:tensorflow:global step 1187: loss = 0.1734 (3.510 sec/step)
INFO:tensorflow:global step 1188: loss = 0.1933 (3.509 sec/step)
INFO:tensorflow:global step 1189: loss = 0.2124 (3.492 sec/step)
INFO:tensorflow:global step 1190: loss = 0.1513 (3.496 sec/step)
```

- Train 104장의 데이터 사용
- Step 1328
- Loss 값이 0.0~0.2에 계속 머무를 때 학습 종료

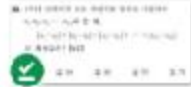
문제 감지 모델 적용 후 예측 box 잘라서 저장



image_11.jpg



image_12.jpg



image_13.jpg



image_21.jpg



image_22.jpg



image_23.jpg



image_24.jpg



image_31.jpg



image_32.jpg



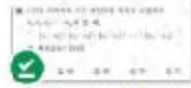
image_33.jpg



image_41.jpg



image_42.jpg



image_43.jpg



image_51.jpg



image_52.jpg



image_61.jpg



image_62.jpg



image_71.jpg



image_72.jpg



image_81.jpg



image_82.jpg



image_91.jpg



image_92.jpg



image_93.jpg



image_94.jpg



image_101.jpg



image_102.jpg



image_103.jpg



image_111.jpg



image_112.jpg



image_121.jpg



image_122.jpg



image_131.jpg



image_132.jpg



image_141.jpg



image_151.jpg



image_152.jpg



image_161.jpg



image_162.jpg



image_163.jpg



image_171.jpg



image_172.jpg



image_173.jpg



image_174.jpg



image_181.jpg



image_182.jpg



image_183.jpg



image_184.jpg



image_191.jpg



image_192.jpg



image_201.jpg



image_202.jpg



image_203.jpg



image_204.jpg



image_211.jpg

항목 감지 모델 학습

```
11206 13:57:19.357709 140621482850176 learning.py:512] global step 726: loss = 0.2164 (2.363 sec/step)
INFO:tensorflow:global step 727: loss = 0.1893 (2.549 sec/step)
11206 13:57:21.908539 140621482850176 learning.py:512] global step 727: loss = 0.1893 (2.549 sec/step)
INFO:tensorflow:global step 728: loss = 0.2277 (2.492 sec/step)
11206 13:57:24.402520 140621482850176 learning.py:512] global step 728: loss = 0.2277 (2.492 sec/step)
INFO:tensorflow:global step 729: loss = 0.3618 (2.342 sec/step)
11206 13:57:26.746000 140621482850176 learning.py:512] global step 729: loss = 0.3618 (2.342 sec/step)
INFO:tensorflow:global step 730: loss = 0.2921 (2.422 sec/step)
11206 13:57:29.169752 140621482850176 learning.py:512] global step 730: loss = 0.2921 (2.422 sec/step)
INFO:tensorflow:global step 731: loss = 0.2464 (2.178 sec/step)
11206 13:57:31.349618 140621482850176 learning.py:512] global step 731: loss = 0.2464 (2.178 sec/step)
INFO:tensorflow:global step 732: loss = 0.5553 (2.455 sec/step)
11206 13:57:33.806173 140621482850176 learning.py:512] global step 732: loss = 0.5553 (2.455 sec/step)
INFO:tensorflow:global step 733: loss = 0.3229 (2.544 sec/step)
11206 13:57:36.352124 140621482850176 learning.py:512] global step 733: loss = 0.3229 (2.544 sec/step)
INFO:tensorflow:global step 734: loss = 0.3906 (2.046 sec/step)
11206 13:57:38.399770 140621482850176 learning.py:512] global step 734: loss = 0.3906 (2.046 sec/step)
INFO:tensorflow:global step 735: loss = 0.2707 (1.765 sec/step)
11206 13:57:40.166721 140621482850176 learning.py:512] global step 735: loss = 0.2707 (1.765 sec/step)
INFO:tensorflow:global step 736: loss = 0.3130 (2.791 sec/step)
11206 13:57:42.959575 140621482850176 learning.py:512] global step 736: loss = 0.3130 (2.791 sec/step)
INFO:tensorflow:global step 737: loss = 0.4943 (2.388 sec/step)
11206 13:57:45.349650 140621482850176 learning.py:512] global step 737: loss = 0.4943 (2.388 sec/step)
INFO:tensorflow:global step 738: loss = 0.2429 (2.375 sec/step)
11206 13:57:47.726332 140621482850176 learning.py:512] global step 738: loss = 0.2429 (2.375 sec/step)
INFO:tensorflow:global step 739: loss = 0.3120 (2.482 sec/step)
11206 13:57:50.218731 140621482850176 learning.py:512] global step 739: loss = 0.3120 (2.482 sec/step)
INFO:tensorflow:global step 740: loss = 0.5997 (2.279 sec/step)
11206 13:57:52.499927 140621482850176 learning.py:512] global step 740: loss = 0.5997 (2.279 sec/step)
INFO:tensorflow:global step 741: loss = 0.2979 (2.448 sec/step)
11206 13:57:54.949341 140621482850176 learning.py:512] global step 741: loss = 0.2979 (2.448 sec/step)
INFO:tensorflow:global step 742: loss = 0.3161 (2.291 sec/step)
11206 13:57:57.242100 140621482850176 learning.py:512] global step 742: loss = 0.3161 (2.291 sec/step)
INFO:tensorflow:global step 743: loss = 0.2069 (2.637 sec/step)
11206 13:57:59.880959 140621482850176 learning.py:512] global step 743: loss = 0.2069 (2.637 sec/step)
INFO:tensorflow:global step 744: loss = 0.3323 (2.169 sec/step)
11206 13:58:02.051901 140621482850176 learning.py:512] global step 744: loss = 0.3323 (2.169 sec/step)
INFO:tensorflow:global step 745: loss = 0.3288 (2.557 sec/step)
11206 13:58:04.610688 140621482850176 learning.py:512] global step 745: loss = 0.3288 (2.557 sec/step)
```

- Train 110장의 데이터 사용
- Step 1097
- Loss 값이 0.2~ 0.3에 계속 머무를 때 학습 종료

배경 분리 Segmentation 모델 학습



Detectron2

```
[4] from detectron2.data.datasets import register_coco_instances
register_coco_instances("test-paper", {}, "./data/trainval.json", "./data/images")
paper_metadata = MetadataCatalog.get("test-paper")
dataset_dicts = DatasetCatalog.get("test-paper")
```

시험지 객체 추가

```
[10/07 12:26:14 d2.engine.train_loop]: Starting training from iteration 0
/usr/local/lib/python3.7/dist-packages/torch/_tensor.py:575: UserWarning: floor_divide is deprecated, and will be removed in a future version of pytorch. It currently rounds toward 0 (like the 'trunc' function NOT 'floor'). This
To keep the current behavior, use torch.div(a, b, rounding_mode='trunc'), or for actual floor division, use torch.div(a, b, rounding_mode='floor'). (Triggered internally at
../aten/src/ATen/native/BinaryOps.cpp:467.)
```

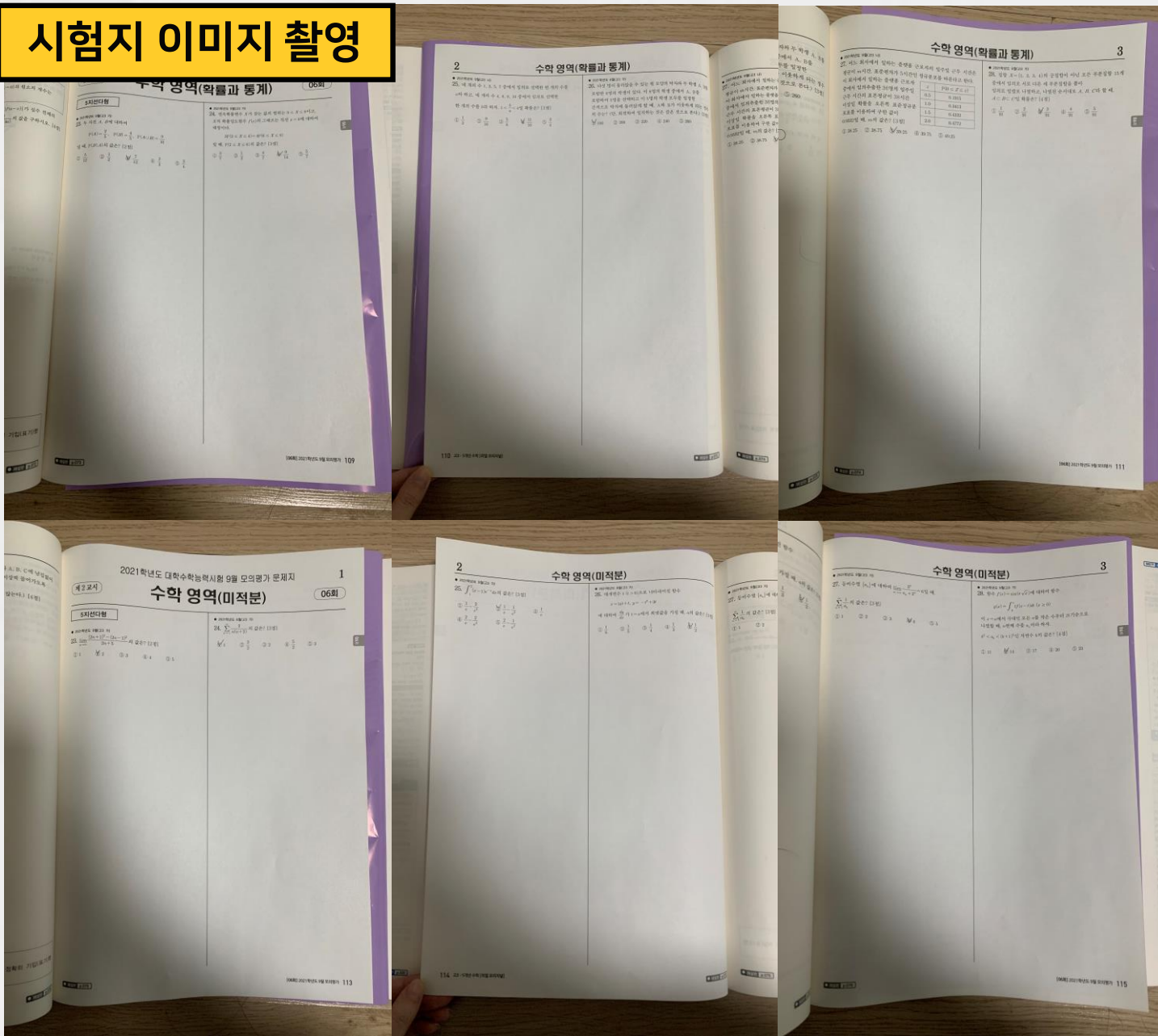
학습 진행

```
return torch.floor_divide(self, other)
[10/07 12:26:42 d2.utils.events]: eta: 0:06:11 iter: 19 total_loss: 0.8225 loss_cls: 0.2126 loss_box_reg: 0.278 loss_mask: 0.3089 loss_rpn_cls: 0.005484 loss_rpn_loc: 0.01969 time: 1.3246 data_time: 0.1266 lr: 0.0012854 max_mem: 2340M
[10/07 12:27:09 d2.utils.events]: eta: 0:05:38 iter: 39 total_loss: 0.3196 loss_cls: 0.02671 loss_box_reg: 0.1795 loss_mask: 0.1104 loss_rpn_cls: 0.003198 loss_rpn_loc: 0.02689 time: 1.3264 data_time: 0.0648 lr: 0.0026174 max_mem: 2340M
[10/07 12:27:35 d2.utils.events]: eta: 0:05:13 iter: 59 total_loss: 0.2054 loss_cls: 0.02559 loss_box_reg: 0.09577 loss_mask: 0.05808 loss_rpn_cls: 0.001786 loss_rpn_loc: 0.02039 time: 1.3233 data_time: 0.0487 lr: 0.0039494 max_mem: 2340M
[10/07 12:28:02 d2.utils.events]: eta: 0:04:49 iter: 79 total_loss: 0.2078 loss_cls: 0.02123 loss_box_reg: 0.09119 loss_mask: 0.04938 loss_rpn_cls: 0.001867 loss_rpn_loc: 0.022 time: 1.3353 data_time: 0.0440 lr: 0.0052814 max_mem: 2340M
[10/07 12:28:29 d2.utils.events]: eta: 0:04:21 iter: 99 total_loss: 0.1862 loss_cls: 0.02193 loss_box_reg: 0.0914 loss_mask: 0.03587 loss_rpn_cls: 0.002019 loss_rpn_loc: 0.02531 time: 1.3312 data_time: 0.0477 lr: 0.0066134 max_mem: 2340M
[10/07 12:28:56 d2.utils.events]: eta: 0:03:55 iter: 119 total_loss: 0.173 loss_cls: 0.02084 loss_box_reg: 0.08955 loss_mask: 0.03835 loss_rpn_cls: 0.002626 loss_rpn_loc: 0.02527 time: 1.3365 data_time: 0.0549 lr: 0.0079454 max_mem: 2340M
[10/07 12:29:23 d2.utils.events]: eta: 0:03:30 iter: 139 total_loss: 0.1754 loss_cls: 0.02026 loss_box_reg: 0.08521 loss_mask: 0.03915 loss_rpn_cls: 0.001633 loss_rpn_loc: 0.02132 time: 1.3373 data_time: 0.1009 lr: 0.0092774 max_mem: 2340M
[10/07 12:29:50 d2.utils.events]: eta: 0:03:06 iter: 159 total_loss: 0.1669 loss_cls: 0.01591 loss_box_reg: 0.08184 loss_mask: 0.03927 loss_rpn_cls: 0.003619 loss_rpn_loc: 0.03095 time: 1.3406 data_time: 0.0390 lr: 0.010609 max_mem: 2340M
[10/07 12:30:17 d2.utils.events]: eta: 0:02:39 iter: 179 total_loss: 0.169 loss_cls: 0.01854 loss_box_reg: 0.0782 loss_mask: 0.0356 loss_rpn_cls: 0.002267 loss_rpn_loc: 0.02597 time: 1.3420 data_time: 0.0531 lr: 0.011941 max_mem: 2340M
[10/07 12:30:44 d2.utils.events]: eta: 0:02:13 iter: 199 total_loss: 0.1628 loss_cls: 0.0163 loss_box_reg: 0.0751 loss_mask: 0.03651 loss_rpn_cls: 0.002467 loss_rpn_loc: 0.0213 time: 1.3418 data_time: 0.0478 lr: 0.013273 max_mem: 2340M
[10/07 12:31:11 d2.utils.events]: eta: 0:01:46 iter: 219 total_loss: 0.2453 loss_cls: 0.0274 loss_box_reg: 0.102 loss_mask: 0.07395 loss_rpn_cls: 0.003168 loss_rpn_loc: 0.0305 time: 1.3423 data_time: 0.0438 lr: 0.014605 max_mem: 2340M
[10/07 12:31:38 d2.utils.events]: eta: 0:01:19 iter: 239 total_loss: 0.2908 loss_cls: 0.03033 loss_box_reg: 0.1298 loss_mask: 0.06746 loss_rpn_cls: 0.00562 loss_rpn_loc: 0.04013 time: 1.3419 data_time: 0.0497 lr: 0.015937 max_mem: 2340M
[10/07 12:32:05 d2.utils.events]: eta: 0:00:53 iter: 259 total_loss: 0.375 loss_cls: 0.03784 loss_box_reg: 0.1771 loss_mask: 0.1024 loss_rpn_cls: 0.008219 loss_rpn_loc: 0.03238 time: 1.3439 data_time: 0.0520 lr: 0.017269 max_mem: 2340M
[10/07 12:32:32 d2.utils.events]: eta: 0:00:26 iter: 279 total_loss: 0.2437 loss_cls: 0.03536 loss_box_reg: 0.1082 loss_mask: 0.077 loss_rpn_cls: 0.004544 loss_rpn_loc: 0.02136 time: 1.3422 data_time: 0.0592 lr: 0.018601 max_mem: 2340M
[10/07 12:33:02 d2.utils.events]: eta: 0:00:00 iter: 299 total_loss: 0.2177 loss_cls: 0.02296 loss_box_reg: 0.1074 loss_mask: 0.05072 loss_rpn_cls: 0.006396 loss_rpn_loc: 0.02216 time: 1.3442 data_time: 0.0477 lr: 0.019933 max_mem: 2340M
[10/07 12:33:02 d2.engine.hooks]: Overall training speed: 298 iterations in 0:06:40 (1.3442 s / it)
[10/07 12:33:02 d2.engine.hooks]: Total training time: 0:06:43 (0:00:03 on hooks)
```


모델 테스트

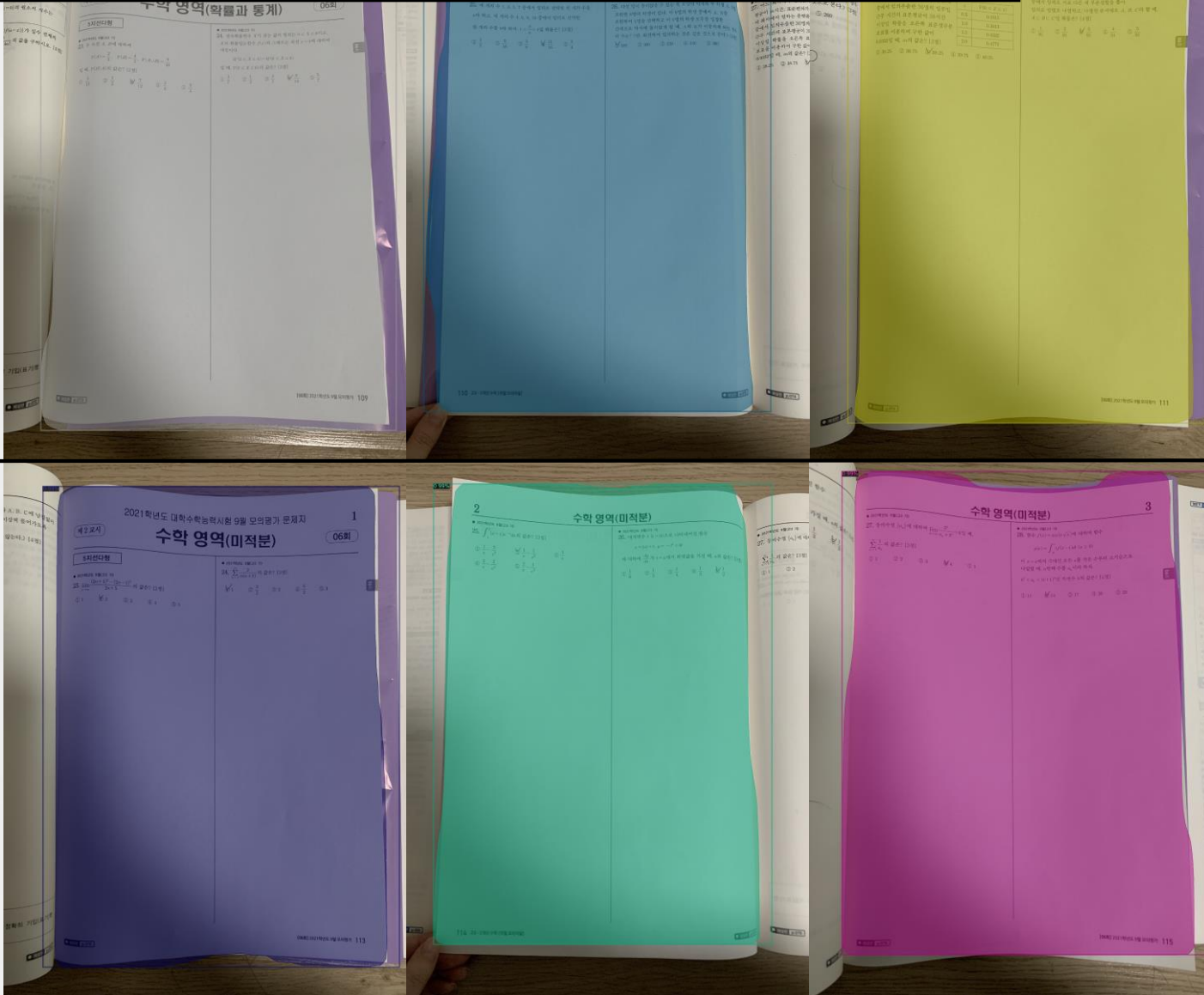
모델 적용과 프로그램 디테일 구현

시험지 이미지 촬영



배경과 분리하는
segmentation 모델

시험지 마스크 영역 색칠 후 출력 + 마스크 행렬 저장

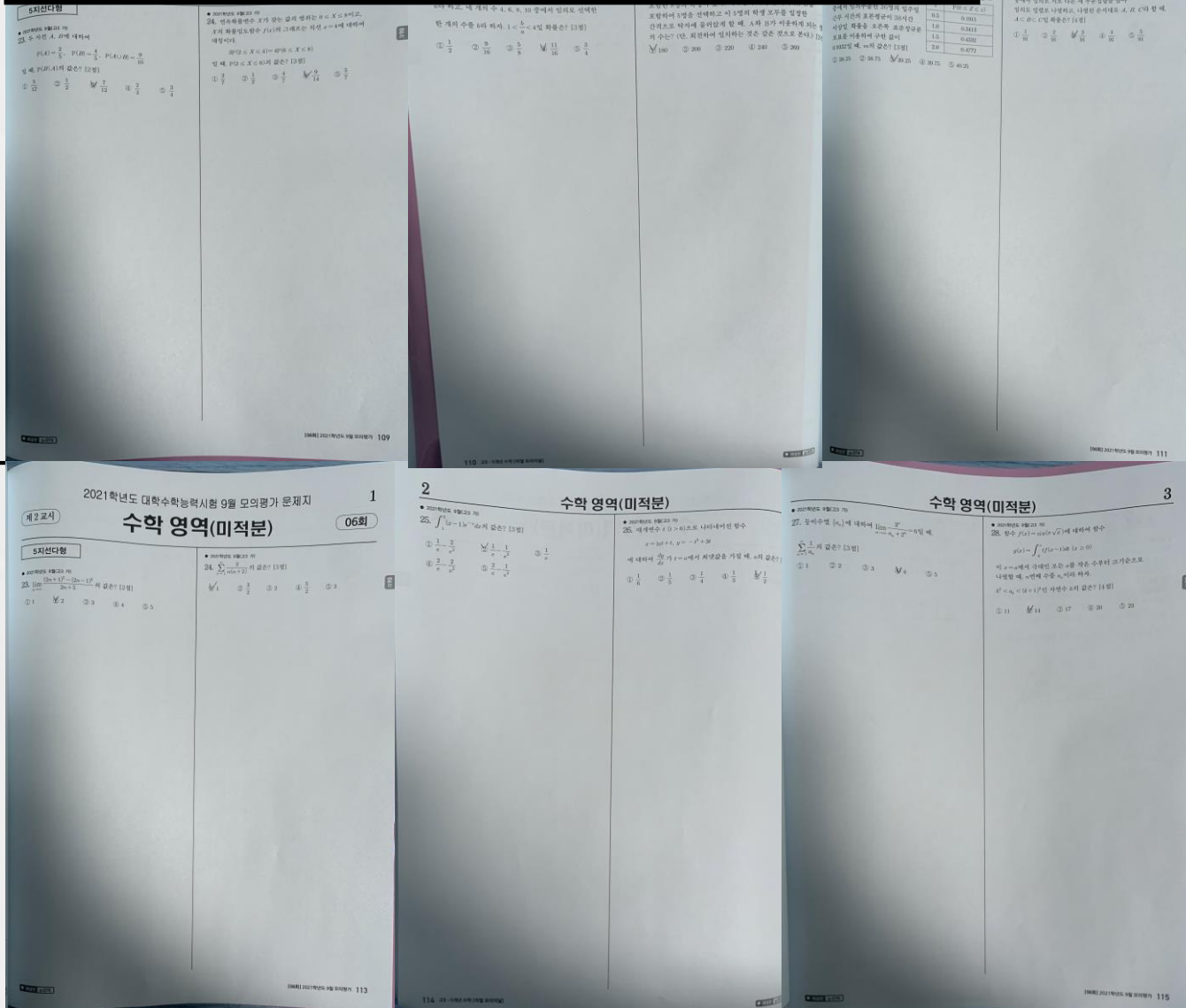


• 시험지 영역 **각 변의 직선의 방정식** 구하기

• 각 직선이 겹쳐지는 **교점** 구하기

• 네 좌표로 **perspective transform** 진행

Perspective Transform 후 자른 시험지 저장



문제 감지 모델

각 페이지 별 문제 잘라 저장

2021학년도 9월(고3 나)

27. 어느 회사에서 일하는 플랫폼 근로자의 일주일 근무 시간은 평균이 m 시간, 표준편차가 5시간인 정규분포를 따른다고 한다. 이 회사에서 일하는 플랫폼 근로자 중에서 임의추출한 36명의 일주일 근무 시간의 표본평균이 38시간 이상일 확률을 오른쪽 표준정규분포표를 이용하여 구한 값이 0.9332일 때, m 의 값은? [3점]

z	$P(0 \leq Z \leq z)$
0.5	0.1915
1.0	0.3413
1.5	0.4332
2.0	0.4772

① 38.25 ② 38.75 ③ 39.25 ④ 39.75 ⑤ 40.25

모델 적용 후
output 딕셔너리의
bounding box
좌표값 활용

항목 감지 모델

문제별 항목 감지

선택항목 : 파란색

선택하지 않은 항목 : 노란색

$$P(A) = \frac{2}{5}, P(B) = \frac{4}{5}, P(A \cup B) = \frac{9}{10}$$

일 때, $P(B|A)$ 의 값은? [2점]

- ① $\frac{5}{12}$
 ② $\frac{1}{2}$
 ③ $\frac{7}{12}$
 ④ $\frac{2}{3}$
 ⑤ $\frac{3}{4}$

● 2021학년도 9월(고3 가)

24. 연속확률변수 X 가 갖는 값의 범위는 $0 \leq X \leq 8$ 이고 X 의 확률밀도함수 $f(x)$ 의 그래프는 직선 $x=4$ 에 대하여 대칭이다.

$$3P(2 \leq X \leq 4) = 4P(6 \leq X \leq 8)$$

일 때, $P(2 \leq X \leq 6)$ 의 값은? [3점]

- ① $\frac{3}{7}$
 ② $\frac{1}{2}$
 ③ $\frac{4}{7}$
 ④ $\frac{9}{14}$
 ⑤ $\frac{5}{7}$

● 2021학년도 9월(고3 나)

3, 5, 7 중에서 임의로 선택한 한 개의 수를 a 라 하고, 네 개의 수 4, 6, 8, 10 중에서 임의로 선택한

한 개의 수를 b 라 하자. $1 < \frac{b}{a} < 4$ 일 확률은? [3점]

- ① $\frac{1}{2}$
 ② $\frac{9}{16}$
 ③ $\frac{5}{8}$
 ④ $\frac{11}{16}$
 ⑤ $\frac{3}{4}$

● 2021학년도 9월(고3 가)

26. 다섯 명이 둘러앉을 수 있는 원 모양의 탁자와 두 학생 A, B를 포함한 8명의 학생이 있다. 이 8명의 학생 중에서 A, B를 포함하여 5명을 선택하고 이 5명의 학생 모두를 일정한 간격으로 탁자에 둘러앉게 할 때, A와 B가 이웃하게 되는 수는? (단, 회전하여 일치하는 것은 같은 것으로 본다.)

- ① 180
 ② 200
 ③ 220
 ④ 240
 ⑤ 260

항목 나열 3가지 패턴에 따른 선택 항목의 번호 판단

1. 한 줄로 배열된 경우

① $\frac{2\sqrt{3}}{9}$ ② $\frac{5\sqrt{3}}{18}$ ③ $\frac{\sqrt{3}}{3}$ ④ $\frac{7\sqrt{3}}{18}$ ⑤ $\frac{4\sqrt{3}}{9}$

2. 두 줄로 배열된 경우

① $\frac{2\sqrt{3}-3}{2}$ ② $\sqrt{3}-1$ ③ $\frac{3\sqrt{3}-3}{2}$
 ④ $\frac{2\sqrt{3}-1}{2}$ ⑤ $\frac{4\sqrt{3}-3}{2}$

3. 세 줄로 배열된 경우

① 0.0228 ② 0.0668
 ③ 0.1587 ④ 0.2417
 ⑤ 0.3085

패턴 별 인식 시각화 및 번호 출력

12. 어느 쌀 모으기 행사에 참여한 각 학생이 기부한 쌀의 무게는 평균이 1.5kg, 표준편차가 0.2kg인 정규분포를 따른다고 한다. 이 행사에 참여한 학생 중 임의로 1명을 선택할 때, 이 학생이 기부한 쌀의 무게가 1.3kg 이상이고 1.8kg 이하일 확률을 오른쪽 표준정규분포표를 이용하여 구한 것은? [3점]

z	P(0 ≤ Z ≤ z)
1.00	0.3413
1.25	0.3944
1.50	0.4332
1.75	0.4599

① 0.6826 ② 0.7357 ③ 0.7745
 ④ 0.8012 ⑤ 0.8543

[[0.8221458, 0.054221
 [[0.8221458, 0.054221
 2 번을 선택했습니다.

최종 출력 화면

1 번 : 3

2 번 : 4

3 번 : 4

4 번 : 1

5 번 : 3

6 번 : 3

7 번 : 2

8 번 : 1

9 번 : 2

10 번 : 5

11 번 : 4

12 번 : 2

5 번의 정답: 2 선택한 답 : 3

11 번의 정답: 3 선택한 답 : 4

12 번의 정답: 1 선택한 답 : 2

번호 별 사용자의 선택 항목 인식 후 출력

정답과 비교 후 틀린 문제 출력

결론

한계점과 느낀점

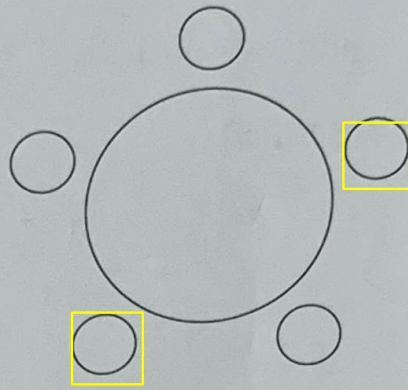
한계점

한계 1. 항목기호와 비슷한 그림이 포함될 때 제대로 감지하지 못함

● 2021학년도 9월(고3 가)

26. 다섯 명이 둘러앉을 수 있는 원 모양의 탁자와 두 학생 A, 포함된 8명의 학생이 있다. 이 8명의 학생 중에서 A, B를 포함하여 5명을 선택하고 이 5명의 학생 모두를 일정한 간격으로 탁자에 둘러앉게 할 때, A와 B가 이웃하게 되는 수는? (단, 회전하여 일치하는 것은 같은 것으로 본다.) [

① 180 ② 200 ③ 220 ④ 240 ⑤ 260



한계점

한계 2. 눈으로 보고 매기는 것 보다 시간이 오래 걸림

- 이미지를 촬영하고, 문제를 잘라 저장하고, 각각 항목을 인식하는 과정이 시간이 많이 소요된다
- 6페이지의 시험을 인식하는 데 약 12분

생각해본 대안

: 시험지를 사진으로 찍지 않고, 카메라에 시험지를 비추는 동시에 번호를 인식할 수 있도록 동영상에 적용할 수 있고, 문제를 자르지 않고도 멀리서 항목을 잘 감지할 수 있도록 구현

느낀점

- 주제의 목표에 맞게 **결과를 내는 것에 집중**했다.
- 그래서 세부적으로 모델에 대한 **evaluation을 진행하지 못했다**는 것이 아쉽다.
- 그러나 목표한 구현은 모두 완료 했으며, 미리 학습된 모델을 가져와 사용했지만 **직접 수집한 데이터**로 라벨링을 진행하고 **Fine Tuning**을 진행했다.
- 또한 모델 환경을 구축하면서 **모델 학습과 추론의 전반적인 내용을 경험**할 수 있었다.

감사합니다

GitHub : <https://github.com/Nayeon12/Scoring-Paper-Tests-with-RCNN>

Blog : https://velog.io/@nayeon_p00/series/2021CapstoneDesign

시연영상 : <https://youtu.be/nVRXpLfXRBO>