

# 수 중 생 물 인 식

## DIVEROID

2022.10

정 승 기  
심 철 우

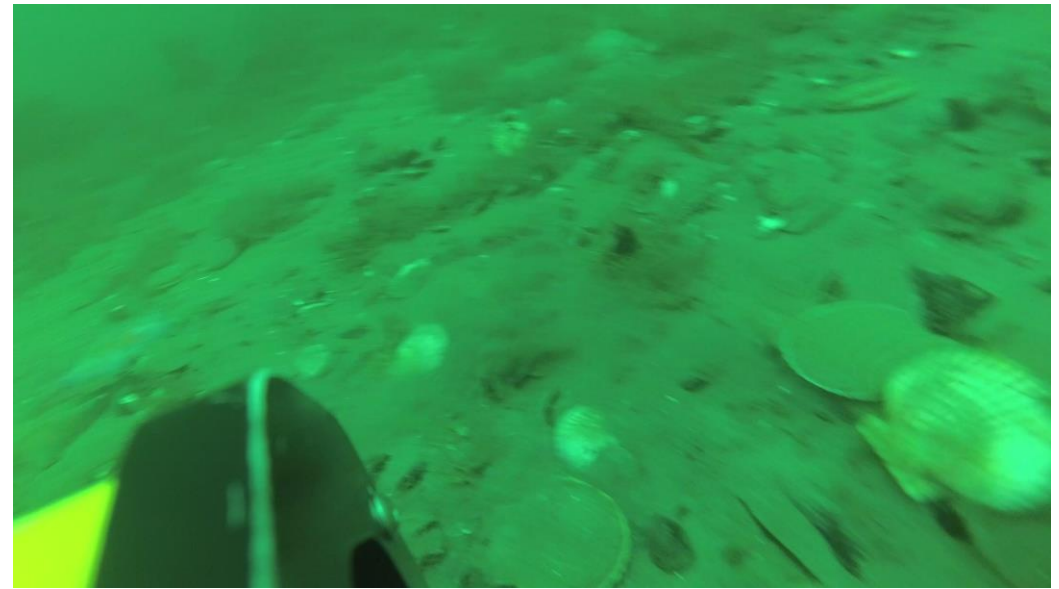
# 목 차 Contents

1. 목 적
2. 파이프 라인
3. 결과 및 테스트
4. 데이터 라벨링
5. 진행과정 및 최종결과

## 1.1 목 적

일반적인 지상에서의 탐지보다 열악한 환경인 수중에서의 객체 탐지 향상

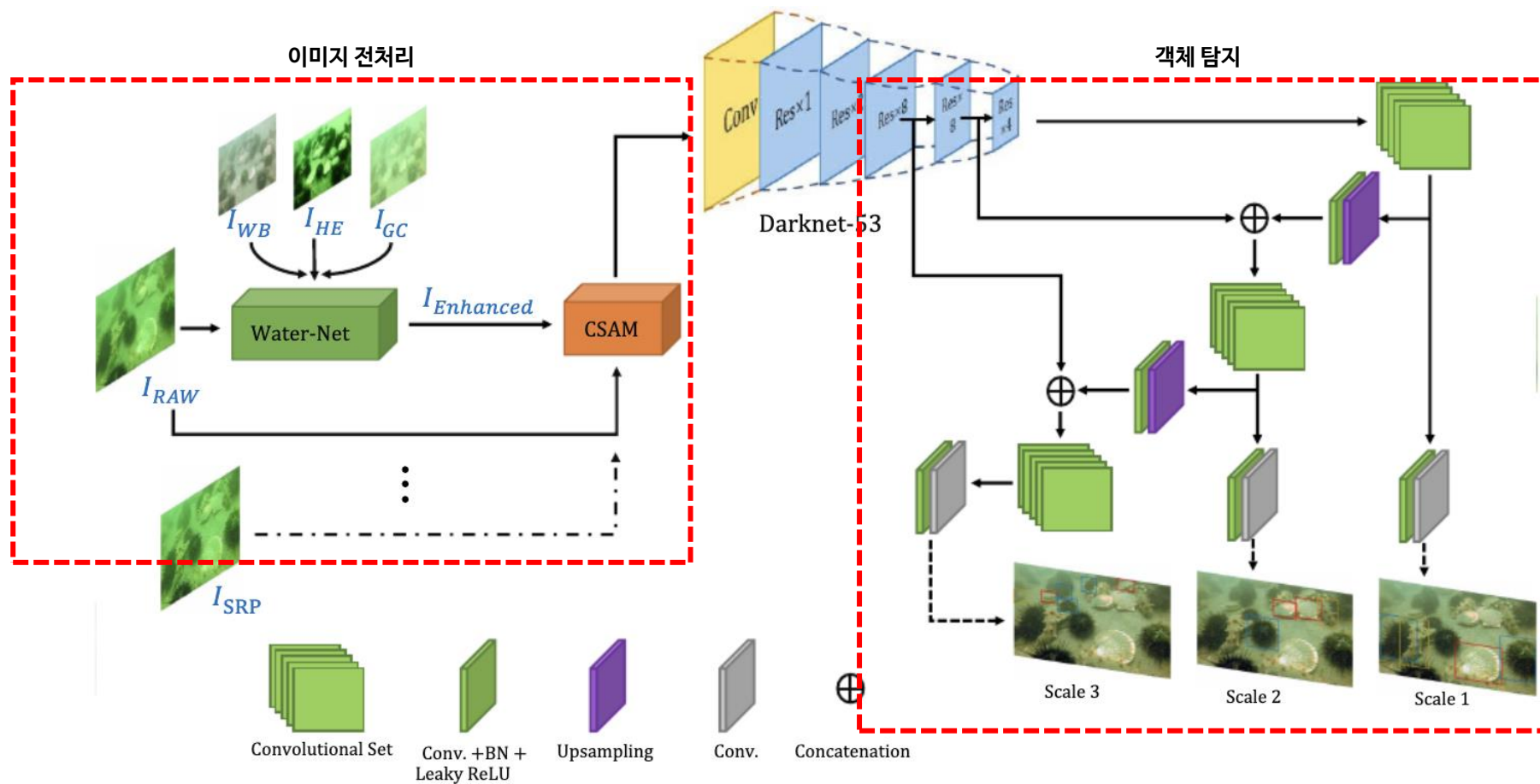
## 1.2 참고 이미지



수중환경에서는 저조도, 고탁도, 낮은 색상대비 등 다양한 요소들을 고려해야 함

# 2 파이프라인

## 2.1 파이프라인



Water-Net, CSAM(Channel Sharpening Attention Module)을 통한 데이터 전처리후 객체 탐지 진행

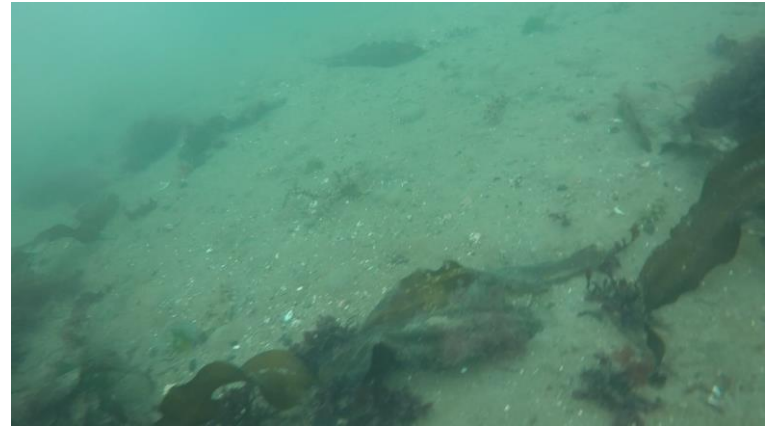
## 2.2 Water-net



적용 전



적용 후



적용 전



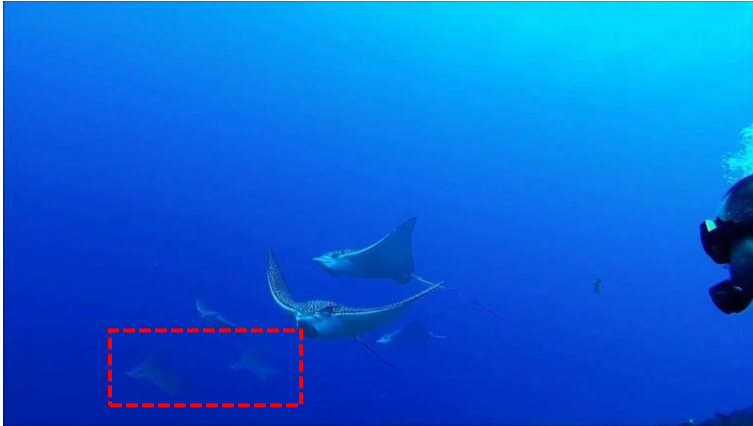
적용 후

Water-Net(사전학습모델)을 통해 이미지의 GC, WB, HE 개선

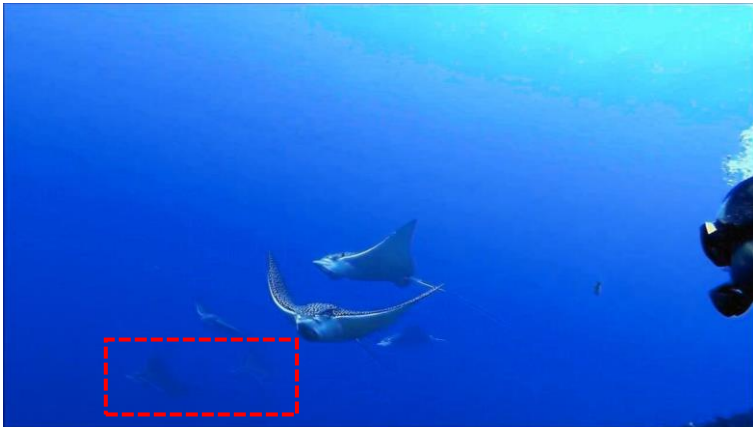


# 2 파이프라인

## 2.2 Water-net



적용 전



적용 후



적용 전



적용 후

문제점도 발견되었는데 센터의 물체는 오히려 대비가 좋아졌지만 뒤에 있는 물체는 오히려 대비가 적어져 분간하기가 어려움  
다른 예시로는 오히려 품질은 더 떨어지는 것으로 보이는 이미지들이 있었음

# 2 파이프라인

## 2.3 CSAM (Channel Sharpening Attention Module)

### 2.3.1 대안 A (OpenCV sharpen)

sharpen



원본이미지



sharpen



원본이미지



OpenCV를 이용하여 sharpen을 주었지만 뭉개지는 경우 발생  
따라서 이미지의 상태에 따라 sharpen의 강도를 다르게 주는 모델이 필요

## 2.3 CSAM(Channel Sharpening Attention Module)

### 2.3.2 대안 B(ESRGAN : Enhanced Super Resolution Generative Adversarial Network)

원본이미지



다운샘플링



ESRGAN



파일 정보

파일 이름  
Original Image (2)

찍은 날짜  
2022년 10월 4일  
오후 11 01

크기  
26.8KB

크기  
416 x 300



파일 정보

파일 이름  
Super Resolution (2)

찍은 날짜  
2022년 10월 4일  
오후 11 01

크기  
210.9KB

크기  
1664 x 1200



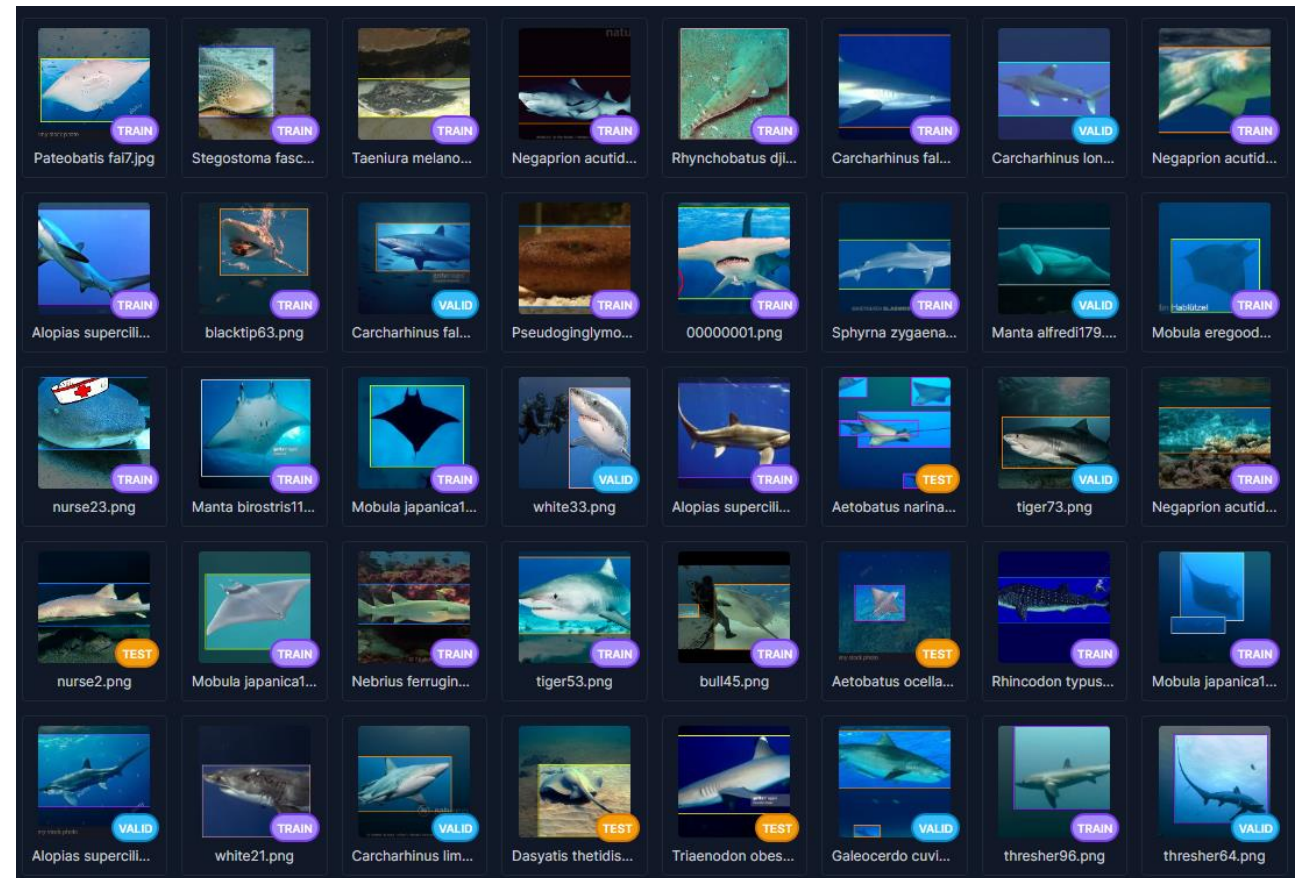
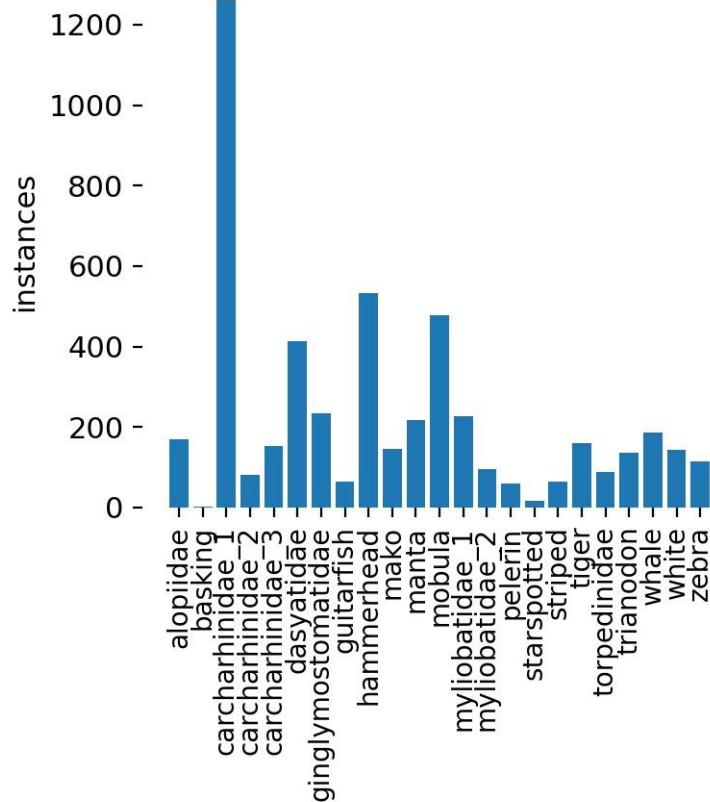
ESRGAN 모델은 128 크기에 효과적이어서 받은 데이터크기 416사이즈에서는 효과가 없음



# 3

## 결과 및 테스트

### 3.1 Result



데이터는 416사이즈로 총 5439개, 23개의 라벨로 구성

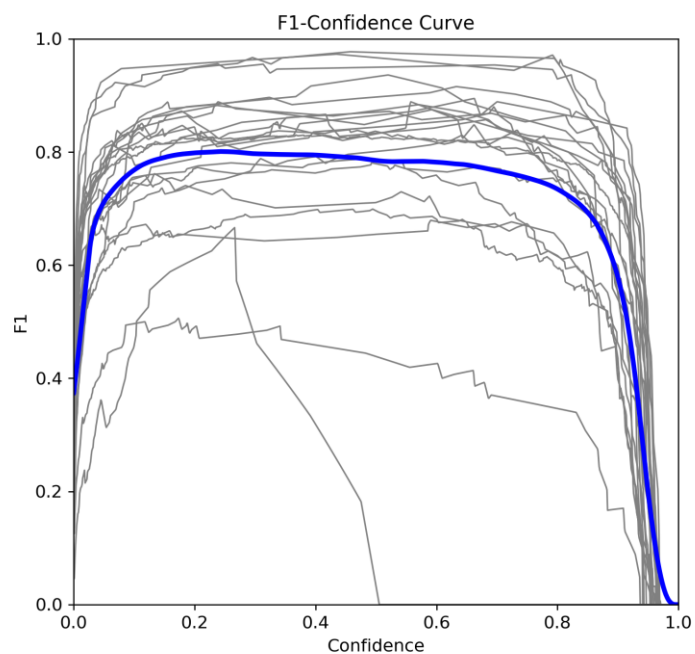
## 3.1 Result(Yolov5s)

Class	Images	Instances	P	R
all	1085	1429	0.806	0.806

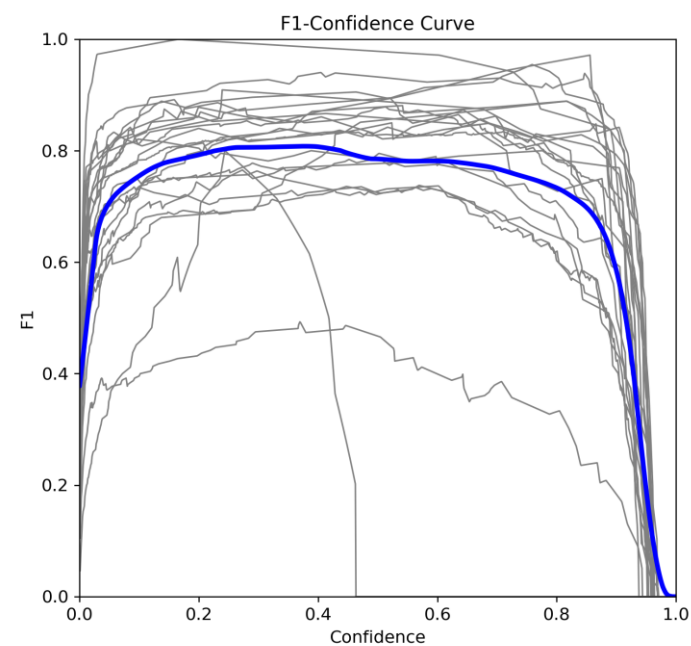
mAP50	mAP50-95
0.838	0.688

Class	Images	Instances	P	R
all	1085	1429	0.842	0.79

mAP50	mAP50-95
0.849	0.695



Water-Net X



Water-Net O

큰 차이는 없지만 이미지 전처리를 한 후 mAP가 조금 더 높은 것을 확인  
또한 F1\_Confidence Curve를 통해 이미지 전처리를 통해 훈련을 한 모델은 더 높은 신뢰도에서 좋은 결과가 나오므로 성능이 더 좋은 것으로 판단

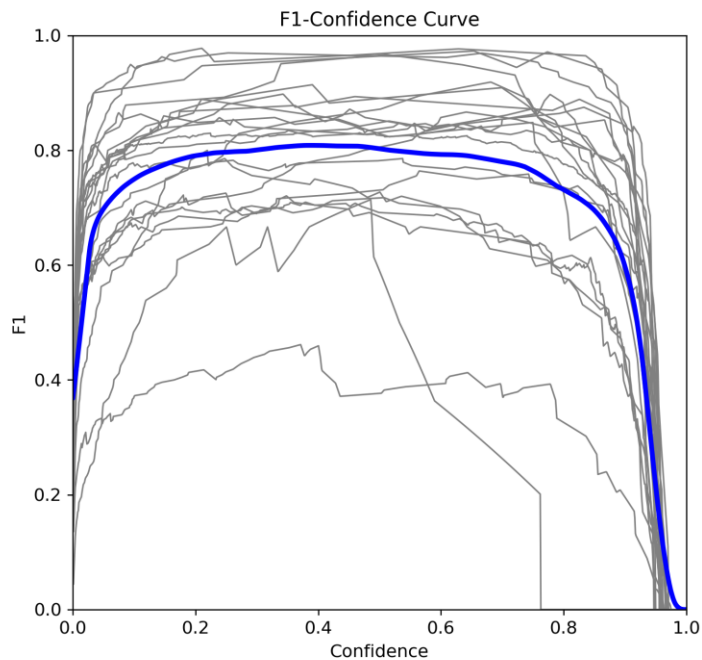
## 3.1 Result(Yolov5s)

Class	Images	Instances	P	R
all	1085	1429	0.845	0.783

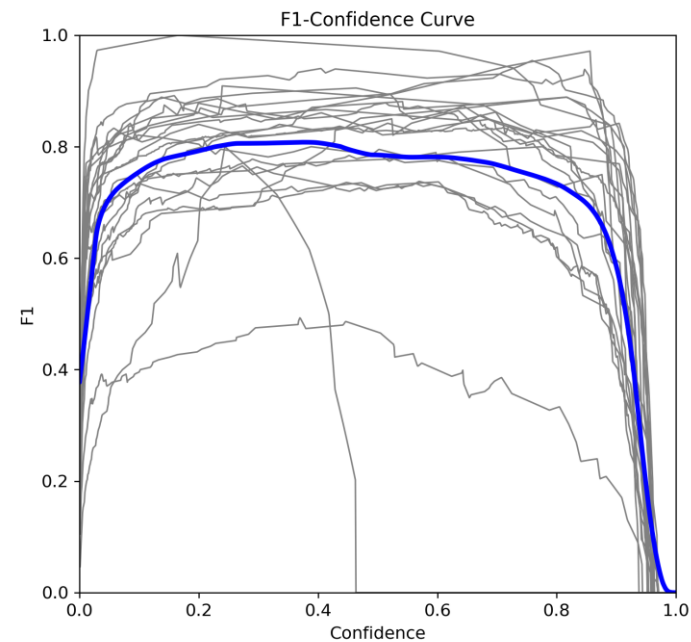
mAP50	mAP50-95
0.836	0.692

Class	Images	Instances	P	R
all	1085	1429	0.842	0.79

mAP50	mAP50-95
0.849	0.695



Water-Net, Sharpen



Water-Net

CSAM을 구현 못해서 샵을 처리해서 직접 테스트 해본 결과 오히려 mAP는 가장 낮게 나옴  
따라서 뒤에 결과는 Water-Net만 적용하여 이미지 전처리 결과

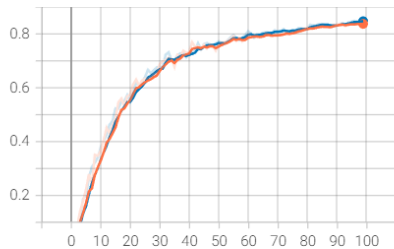
## 3

## 결과 및 테스트

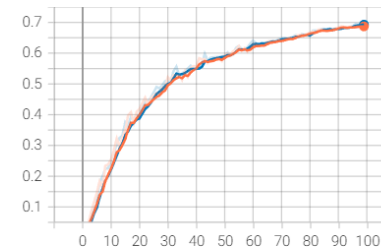
## 3.1 Result

— 이미지 전처리 O  
— 이미지 전처리 X

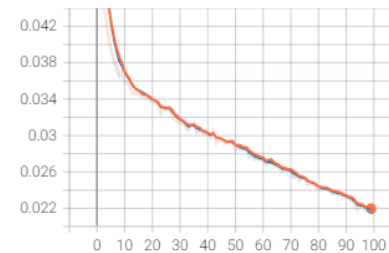
metrics/mAP\_0.5  
tag: metrics/mAP\_0.5



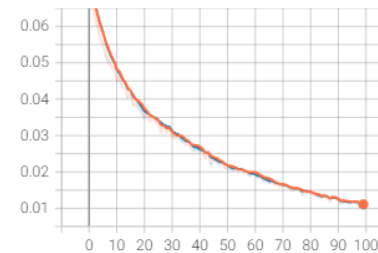
metrics/mAP\_0.5:0.95  
tag: metrics/mAP\_0.5:0.95



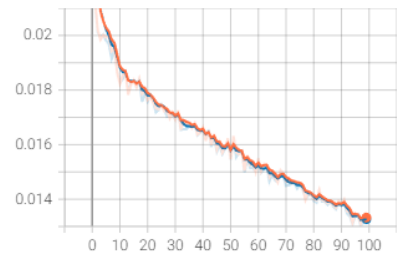
train/box\_loss  
tag: train/box\_loss



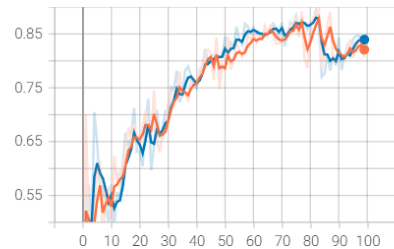
train/cls\_loss  
tag: train/cls\_loss



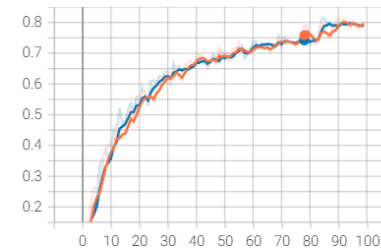
train/obj\_loss  
tag: train/obj\_loss



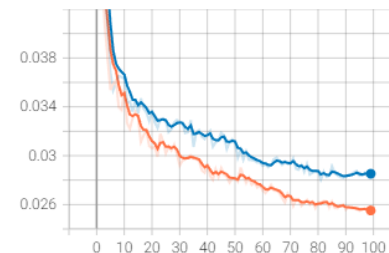
metrics/precision  
tag: metrics/precision



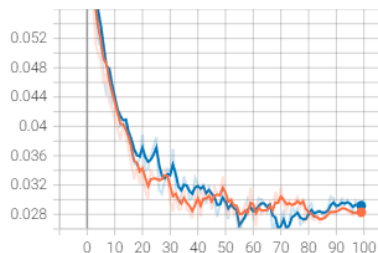
metrics/recall  
tag: metrics/recall



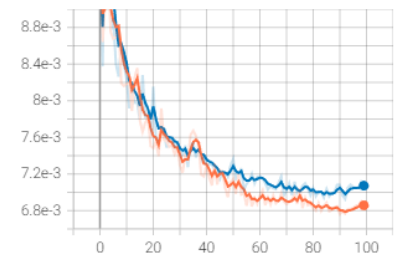
val/box\_loss  
tag: val/box\_loss



val/cls\_loss  
tag: val/cls\_loss



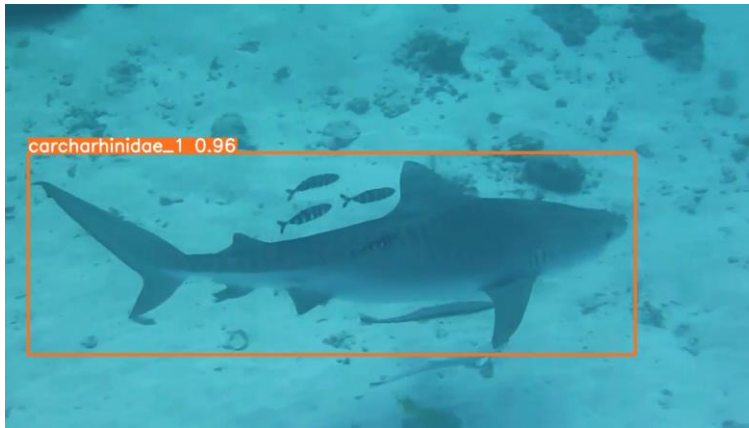
val/obj\_loss  
tag: val/obj\_loss



다른 그래프들은 유사하지만 val데이터에 대해 box\_loss, obj\_loss 의 경우 손실이 더 발생  
조도, 색상 등 더 좋아짐으로 탐지 범위가 더 늘어난 것으로 추측



## 3.2 Test\_Case1



적용 전



적용 후



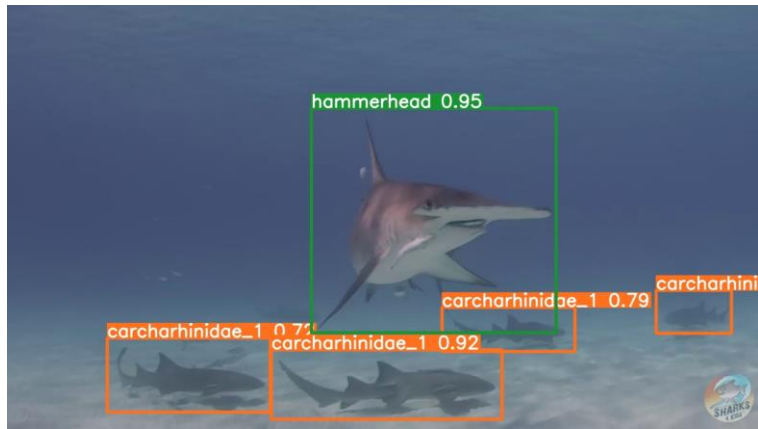
적용 전



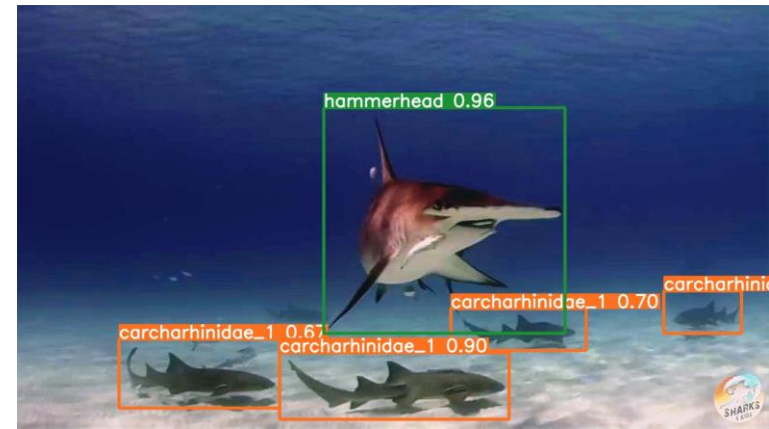
적용 후

데이터 전처리 적용 전과 후의 결과가 다른 경우로 **적용 후 결과가 더 정확한 것을 확인**

## 3.2 Test\_Case2



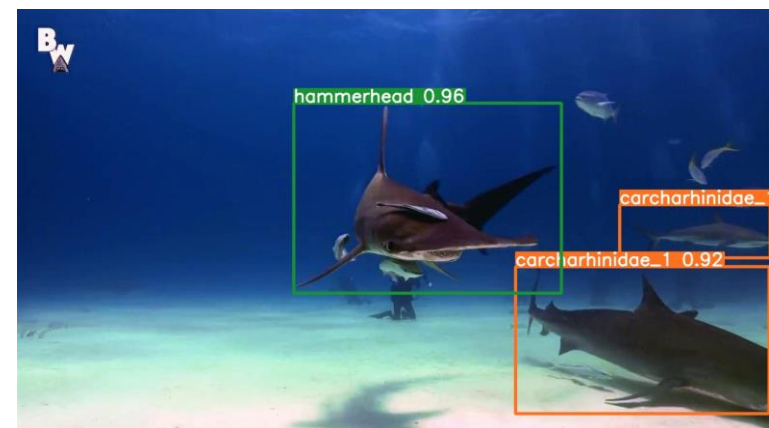
적용 전



적용 후



적용 전



적용 후

같은 결과라도 큰 차이는 없지만 중심이 되는 물체는 신뢰도가 높아졌지만 주변의 물체는 떨어지는 것을 확인  
앞서 말한 water-net의 문제점 때문으로 예상

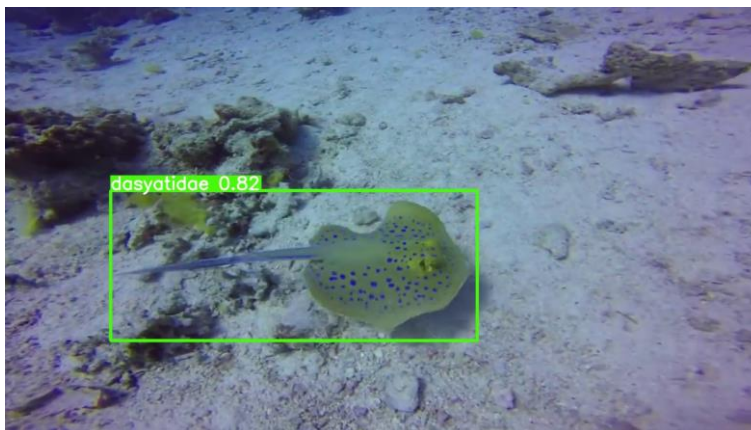
## 3.2 Test\_Case3



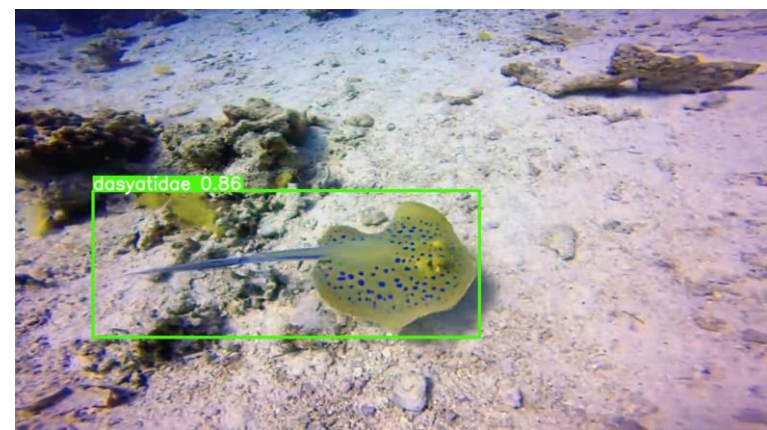
적용 전



적용 후



적용 전



적용 후

결과는 같으나 신뢰도가 더 떨어지거나 박스가 더 커지는 경우  
이런 경우 때문에 결과 그래프에서 val/box loss가 더 큰 것으로 추측



# 4 데이터 라벨링

## 4.1 데이터 라벨링



### MakeSense

- 혼자사용하고 적은데이터엔 편함
- 중간 저장없음
- 원하는 타입(json,xml,txt)으로 변환 가능
- 공유 안됨



### Roboflow

- 이미지증강사용하기엔 편함
- 데이터 리사이즈 편함
- 원하는 타입(json,xml,txt)으로 변환 가능
- 공유 가능



### Supervisely

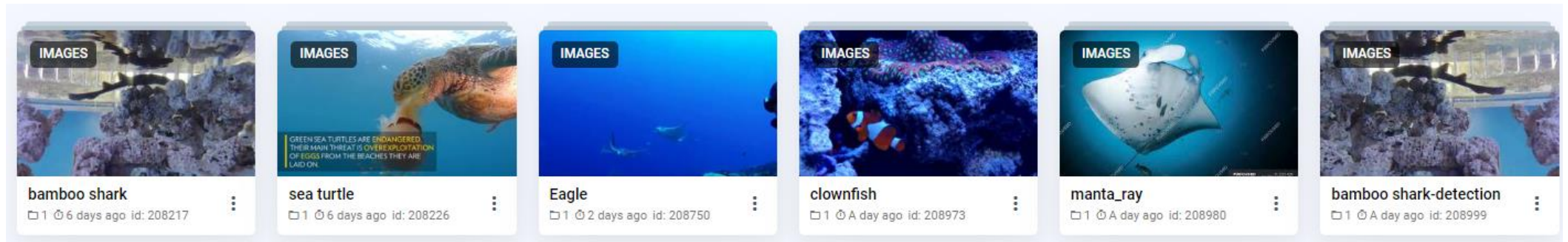
- 많은 데이터들 사용할 때 편함(단축키 사용)
- 공유 가능
- 폴더관리 편함

모두 장단점이 있어 상황에 맞게 사용하기로 함



# 4 데이터 라벨링

## 4.1 데이터 라벨링



Supervisely 예시



카테고리별 라벨링 작업



라벨링 취합

한 카테고리마다 작업 시 라벨을 변경 필요없어 시간이 효율적이지만 txt로 받을 시 모두 클래스가 0번으로 표기  
따라서 json파일로 받아 roboflow에 업로드하여 txt로 받을 시 클래스가 잘 나누어짐

## 5.1 진행과정

	1주차 (9/19~9/24)	2주차(9/26~10/2)	3주차 (10/3~10/9)	4주차(10/10~10/14)
	목표 : 데이터 수집	목표: 수중생물 인식 AI 모델링 구현	목표: 모델링 성능 개선 데이터 수집	목표: 최종검토
월	- 데이터 수집 알고리즘 제작	- 모델 파이프라인 구성 - Water-net, CSAM 스테 디	- 개천절	- 대체휴무
화	- 데이터 수집 알고리즘 제작 - 수중생물 50종 선정 - 데이터 수집 검수	- Water-net 구현 및 데이 터 전처리 - 객체 탐지 모델 스테디	- yolov5, yolov7 테 스트	- 데이터 라벨링
수	- 데이터 수집, 검수	- CSAM 외에 다른 모듈 검색 및 스테디	- 데이터 정리 - 이미지 전처리 방 법 스테디	Dead Line
목	- 수중생물 50종 선정 수정 - 데이터 수집, 검수	- yolov5 백본, 헤드 수정 테스트	- OpenCV sharpen 테스트	- 최종 보고서 제출
금	- 데이터 수집, 검수	- Faster-RCNN, SSD 스테 디 및 학습	- 데이터 라벨링	- 최종 결과를 제출
토				
일				

## 5.2 진행사항

## 5.2.1 데이터 수집

	구글이미지	유튜브
수집 카테고리 수	43종	34종
정리된 카테고리 수	43종	23종
라벨링	8종	

## 5.2.2 수중 생물 인식 모델

Water-net(이미지 전처리)	CSAM(이미지 전처리)	객체탐지모델
코드구현(사전학습모델)	X	Yolov5s, Yolov7

Class	Images	Labels	P	R	mAP@0.5	mAP@0.5:0.95
all	1085	1429	0.897	0.801	0.874	0.743

Yolov5s보다 Yolov7을 사용하면 기존 0.849, 0.695에서 항상  
최종적으로 CSAM을 구현하거나 다른 대안방법을 통해 이미지 전처리를 한다면 결과가 더 좋아질 것

