Journal of Korea Robotics Society (2019) 14(2):139-149
https://doi.org/10.7746/jkros.2019.14.2.139 ISSN: 1975-6291 / eISSN: 2287-3961

강건한 CNN기반 수중 물체 인식을 위한 이미지 합성과 자동화된 Annotation Tool

Synthesizing Image and Automated Annotation Tool for CNN based Under Water Object Detection

전 명 환[1] · 이 영 준[2] · 신 영 식[3] · 장 혜 수[4] · 여 태 경[5] · 김 아 영†
MyungHwan Jeon[1], Yeongjun Lee[2], Young-Sik Shin[3], Hyesu Jang[4], Taekyeong Yeu[5], Ayoung Kim†

Abstract: In this paper, we present auto-annotation tool and synthetic dataset using 3D CAD model for deep learning based object detection. To be used as training data for deep learning methods, class, segmentation, bounding-box, contour, and pose annotations of the object are needed. We propose an automated annotation tool and synthetic image generation. Our resulting synthetic dataset reflects occlusion between objects and applicable for both underwater and in-air environments. To verify our synthetic dataset, we use MASK R-CNN as a state-of-the-art method among object detection model using deep learning. For experiment, we make the experimental environment reflecting the actual underwater environment. We show that object detection model trained via our dataset show significantly accurate results and robustness for the underwater environment. Lastly, we verify that our synthetic dataset is suitable for deep learning model for the underwater environments.

Keywords: Deep Learning, Data Annotation, Object Detection, 3D CAD Model

1. 서 론

수중 재난 상황에 이용할 수 있는 로봇은 크게 두 종류로 AUV (Autonomous Underwater Vehicle)와 ROV (Remotely Operated Vehicle)로 분류된다. 이중 AUV는 로봇에 장착된 여러 센서들을 통해 주변 상황을 스스로 인지하고 필요한 작업을 자율적으로 실행하는 더욱 지능화된 로봇을 지칭한다. 센서 정보를 사용하여, 수중 환경에서 목표로 하는 대상을 인지하고 분별해 내는 작업은 다양한 로봇 분야에서 매우 중요하며, 그 과정에서 높은 정확성을 필요로 한다. 최근 대두되고 있는 Deep Learning 기반의 방법들은 환경 변화에 강건한 분별 능력을 보여주고 있다. 하지만 기존 연구 결과는 대부분 지상 환경의 영상에 치중되어 있으며, 수중 환경을 대표하는 데이터가 부족하여, 그로 인해 수중 환경에서의 인공지능 기법은 상대적으로 적게 발표되었다. 구체적으로는, 수중이라는 특수한 환경을 반영한 데이터 셋이 턱없이 부족할 뿐만 아니라, 수중 환경에서 얻어진 정보는 색상 왜곡, Intensity Degeneration과 같은 현상을 가지기 때문에 로봇이 스스로 대상을 분별해 내어 높은 정확성을 가지기가 어렵다. 따라서, 3D CAD 모델을 이용하여 합성 이미지를 만들고, 이 합성 이미지를 렌더링할 때에 수중의 다양한 광학적 조건들을 반영한 데이터 셋을 로봇에 제공해 줄 수 있다면 로봇의 구조 대상 분별 능력을 크게 향상시킬 수 있다.

Received : Jan. 8. 2019; Revised : Mar. 5. 2019; Accepted : Mar. 14. 2019
※ This study is a part of the results of R&D project, Development of Basic Technologies of 3D Object Reconstruction and Robot-Manipulator Motion Compensation Control, supported by KRISO(Korea Research Institute of Ships and Ocean Engineering).
1. Master student, Robotics Program, KAIST, Daejeon, Korea (myunghwan.jeon@kaist.ac.kr)
2. Junior engineer, Korea Research Institute Ship and Ocean engineering (KRISO), Daejeon, Korea (leeyeongjun@kriso.re.kr)
3. Ph.D. student, Dept. of Civil and Environmental Engineering, KAIST, Daejeon, Korea (youngsik.shin@kaist.ac.kr)
4. Master student, Dept. of Civil and Environmental Engineering, KAIST, Daejeon, Korea (iriter@kaist.ac.kr)
5. Senior researcher, Korea Research Institute Ship and Ocean engineering (KRISO), Daejeon, Korea (yeutk@kriso.re.kr)
† Assistant professor, Corresponding author: Dept. of Civil and Environmental Engineering, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon, Korea (ayoungk@kaist.ac.kr)

이러한 데이터 셋의 구축과 학습은 AUV뿐 아니라, 컨트롤러를 통해 사람이 직접 개입하여 원격으로 제어하는 방식인 ROV의 경우에도 적용된다. 수중 환경을 반영한 데이터 셋을 통해 향상된 구조 대상 검출 결과가 ROV를 제어하는 사람에게 주어진다면, 제어하는 사람은 더 좋은 판단을 내릴 수 있다.

따라서 본 연구에서는 CNN(Convolutional Neural Network)를 사용하여 Underwater Object Detection과 Object Pose Estimation에 필요한 데이터 셋을 만드는 방법을 제안한다. 본 연구는 대상과 일치하는 3D CAD 모델과 다양한 광원을 활용하여 수중 환경의 특수한 광학적 조건을 최대한 충족시키는 데이터 셋을 만들어 낸다. 또한 Object Detection과 Object Pose Estimation을 위한 Annotation을 자동으로 생성하는 툴을 제안한다. 최종적으로 CNN 기반 Object Detector에 본 연구에서 만들어낸 데이터 셋을 적용한 실험을 통해 제안하는 데이터 셋의 적합성을 검증한다. 이를 통해 CNN을 사용한 Underwater Object Detection 및 Pose estimation에 다양하고 풍족한 데이터를 제공하여, 더 좋은 결과를 얻을 수 있을 것이라 기대한다. 본 연구는 다음을 제안한다.

- 단일 물체만 다룬 것이 아니라 현실에서 주로 발생하는 다중 물체의 겹침 현상(Occlusion)을 반영한 데이터 셋을 제공한다.
- 기존의 방법이 지상에서의 물체 인식을 위한 인공지능 학습에 집중하였다면, 본 논문은 일반적인 환경뿐만 아니라 수중 환경에서도 적합한 데이터 셋을 구성하는 방법을 구체적으로 설명하고 나아가 수중 로봇에 적용할 수 있는 데이터를 제공한다.
- 최종적으로 본 연구는 Object Detection과 Pose Estimation을 위한 데이터에 집중하며, 특히 해당 학습 데이터의 Annotation을 자동으로 생성하는 툴을 제공한다.

2. 선행 연구 조사

Deep Learning 기반 방식을 사용하기 위해서는 충분한 트레이닝 셋을 확보하는 것이 필수적이다. 그러나 실제로 다양한 데이터 셋을 확보하는 것이 어려운 상황이 빈번하게 발생하기 때문에, 합성 이미지(Synthetic Image)를 만들어 트레이닝 셋을 다양화하는 연구가 많이 행해져 왔다.

2.1 CNN기반 Object Detection

CNN은 주로 Fully Connected Layer 앞에 Convolutional Layer가 여러 개 놓여 있는 형태를 가진다. Convolutional Layer는 입력 이미지의 특징(Feature)들을 추출하고, 이 추출된 특징들을 이용하여 Fully Connected Layer에서 다양한 분류를 해낸다. 최근 CNN을 사용하여 Object Detection을 수행하려는 연구가 많이 진행되고 있다. [1]은 입력 이미지 보다 작은 윈도우를 일정한 간격으로 움직여 입력 이미지를 전부 훑어보는 슬라이딩 윈도우 검색을 사용한다. 이 과정에서 생성된 수많은 패치에서 CNN을 실행하는 것은 어려움이 있었으나, 저자는 주변 픽셀들의 유사도를 검사하여 Grouping 함으로써 객체 검출이 가능한 후보 영역을 찾아내는 방법인 Selective Search[2]를 통해 사용자에게 제공되는 윈도우의 개수를 2000개 정도로 줄여 Object Detection에 CNN을 사용할 수 있게 하였다. 하지만 약 2000개의 윈도우에서 CNN을 실행하는 것은 매우 오래 걸린다는 단점이 있다. [3]은 각 이미지를 여러 개의 그리드로 분할하고, CNN을 통해 그리드 내 객체 인식 정확성을 반영한 신뢰도를 계산하여 가장 높은 객체 인식 정확성을 가지는 경계 상자를 얻는 방식으로 물체를 인식한다. [3]의 처리 과정이 단순하여 속도는 빠르지만 객체의 크기가 작으면 높은 정확성을 기대하기 어렵다. 본 연구에서는 [1]의 향상된 버전인 Mask R-CNN[4]를 활용하여, 제안하는 학습 방법의 유효성을 검증하고자 한다. 기존 Object Detection에서 널리 쓰이는 Faster R-CNN[5]는 Detection 결과인 물체의 Class와 Bounding-Box만 제공한다. 그러나 Mask R-CNN은 추가적으로 Bounding-box내에 있는 Instance의 Mask를 제공하여 추후 데이터 활용을 더 용이하게 한다. 구체적으로는 본 연구에서 만들어 낸 데이터 셋을 사용하여 Mask R-CNN으로 학습시킨 후 Object Detection 성능을 검토하여, 제안하는 학습 방법의 유효성을 검증한다.

2.2 3D CAD 모델을 사용한 이미지 생성

[6, 7]에서는 이미지를 합성하기 위해 3D CAD 모델을 사용하였다. [6]은 3D Ware House[8]에서 20개의 카테고리를 선택하여 카테고리당 25개의 모델을 사용하였다. 이 모델들에 저자가 직접 선택한 색상과 텍스처를 입혀 주었고, 카메라에 대한 물체의 포즈 또한 저자가 직접 바꿔 준 후 렌더링을 통해 이미지를 생성하였다. [7]은 PASCAL 3D+에서 12개의 카테고리에 해당하는 모델들을 사용하였다. 저자는 해당 모델들에 대하여 Viewpoint, 이미지의 배경, 광학적 조건을 무작위로 선택하여 이미지를 생성하였다.

본 연구는 [7]의 방법을 기반으로 수행되었다. 하지만, 하나의 물체만 고려한 [7]과 달리, 여러 개의 물체를 사용하여 물체들 간의 겹침 현상(Occlusion)을 추가로 고려하였고 수중 환경을 반영하기 위해 이미지 hazing을 반영하였다.

2.3 Auto Annotation Tool

실제 이미지에 나타나는 객체들에 자동으로 라벨을 생성하는 것은 매우 힘들기 때문에, 주로 사람이 직접 작업을 수행하는 것이 일반적이다. 그러나 사람이 직접 라벨을 표시하는 것

또한 부정확할 수 있고, 데이터 셋의 양이 많을 경우에는 어려움이 있어 적합하지 못한 방법이다. 이를 해결하기 위해 [9]는 높은 현실감을 반영한 시뮬레이션 엔진을 통해 Deep Learning에 필요한 트레이닝 셋을 만들어 내었다. 시뮬레이션 엔진을 통해 날씨의 변화, 낮과 밤 등의 시간 변화를 주어 트레이닝 셋의 다양성을 확보했고, 또한 이미지의 깊이 정보도 얻어내었다. 또 다른 방법으로는 3D CAD 모델을 사용하여 이미지를 생성한 후 자동으로 라벨을 생성하는 툴에 관한 연구가 행해져 왔다. [7, 10, 11, 12]은 구면 좌표계의 중심에 하나의 물체를 두고, 구면 좌표계의 성분인 Azimuth, Elevation에 대한 샘플을 생성하여 가상 카메라를 이 샘플에 맞게 구면 좌표계에 위치시킨 후, 가상 카메라를 이용하여 물체에 대한 이미지를 촬영한다. 그리고 촬영한 이미지에서 물체의 바운딩 박스와 세그멘테이션 라벨을 추출해 낸다. 본 연구에서는 여러 개의 물체를 사용하여 물체들 간의 겹침 현상을 고려하기 위해, 카메라를 직교 좌표계의 중심에 위치시킨다. 이때 가상 카메라의 FOV 안에서 물체들을 이동시켜 이미지를 촬영한다. 자세한 내용은 3.1.2과 3.1.3에서 설명한다.

3. Auto Annotation tool

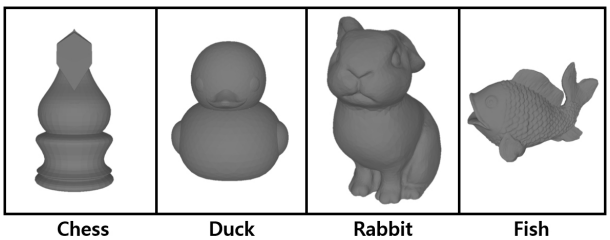
본 연구에서는 3D CAD 모델을 사용하여 이미지를 합성하고 합성한 이미지에 필요한 라벨을 자동으로 생성하는 툴을 제안한다.

3.1 Sample 생성

3.1.1 3D CAD 모델

앞선 연구[7, 10, 11, 12]에서는 주로 PASCAL 3D+, 3D Ware House와 같이, 이용 가능한 물체의 카테고리가 제한된 모델들을 사용하였다. 이와는 달리, 본 연구에서는 온라인에서 검색을 통해 쉽게 찾을 수 있는 모델들을 사용함으로써, 이미지 합성 시 쓰일 수 있는 3D CAD 모델에 어떠한 제한도 두지 않았다. [Fig. 1]에는 본 연구에서 선택한 3D CAD 모델들 중 일부를 표시해 두었다.

본 연구에서 사용한 샘플들은 수중 환경에서 발견할 수 있는 전형적인 물체가 아니며 오히려 더욱 복잡한 형태로 볼 수 있다. 하지만 본 연구는 사용한 네 가지 샘플에 국한되는 것이 아니며, CAD 모델을 가지고 있는 경우 이를 학습함으로써 수중 환경에서의 인식 능력을 향상시킬 수 있음을 증명하는 것을 주요 목표로 한다. 즉, 선택한 모델은 실제 수중에서 쉽게 발견할 수 있는 물체(파이프, 수중 구조물 등)는 아니지만, 이를 통해 임의의 모양을 갖는 물체에 대한 알고리즘의 학습 능력을 검증할 수 있다.



[Fig. 1] Selected 3D CAD model. we select various 3D CAD model to make synthetic image dataset. This figure includes part of selected model.

3.1.2 Virtual Camera

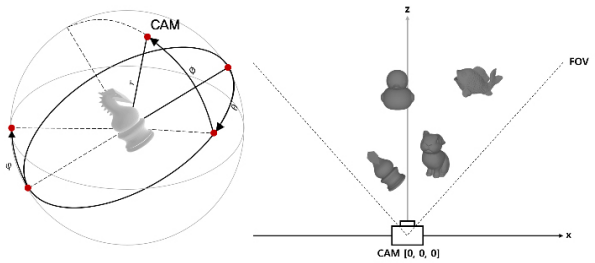
3D CAD 모델로부터 이미지를 렌더링하여 합성 이미지를 만들기 전에 Object Detection과 Pose Estimation에 필요한 Annotation을 먼저 추출한다. 이를 위하여 Virtual Camera를 사용하여, 실제 렌더링된 이미지와는 색상을 제외한 다른 것들은 모두 동일한 이미지를 얻어낸다. 이 이미지는 Object Detection과 Pose Estimation에 필요한 Annotation을 추출하기 위한 기반이 된다.

본 연구는 Blender[13]를 사용하여 3D CAD 모델을 렌더링하고 합성 이미지를 만들어 낸다. 정확한 Annotation 추출을 위해서는 Blender를 통해 생성된 이미지와 Virtual Camera를 통해 만들어진 이미지가 완전히 동일해야 하기에, 실험에서 사용하는 실제 카메라가 아닌 Blender에서 렌더링을 할 때에 사용되는 카메라와 Virtual Camera가 동일한 카메라 내부 파라미터(Intrinsic Parameters)를 갖게 해야 한다. 두 개의 카메라 모두 960×540 해상도를 가지는 영상으로 정의하였으며, 해당 이미지 생성을 위한 카메라의 내부 파라미터는 식 (1)의 값으로 정한다. 또한, Virtual Camera의 FOV안에 물체가 없을 때에는 RGB의 값이 모두 255인 하얀색의 이미지를 얻을 수 있게 만들어 주었다.

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} 1120 & 0 & 480 \\ 0 & 1120 & 270 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

3.1.3 물체의 포즈 샘플 생성

앞선 연구[7]에서는 [Fig. 2]의 (a)와 같이 구면 좌표계를 활용하였다. 이 연구에서는 구면 좌표계의 중심에 물체를 위치시키기 때문에 다중 물체들 간의 올바른 겹침 현상(Occlusion)을 반영하기 힘들다. 본 연구에서는 다중 물체를 고려하여 물체들 간의 Occlusion을 고려하기 위해 카메라는 직교 좌표계의 중심인 (0, 0, 0)에 위치시킨다. 그리고 카메라의 FOV안에서 물체들을 이동시키게 된다([Fig. 2(b)]). 이때 카메라와 물체 간의 Relative Transformation을 임의로 만들어 주어야 한다. 샘플링해야 할 파라미터는 총 6개로 Translation 속성인 X, Y, Z와 회전 속성인 Roll, Pitch, Yaw이다. 회전 속성인 Roll, Pitch, Yaw는 정해진 범위(0°~360°) 안에서 무작위로 생성되며, Translation 속성인 X, Y, Z는 카메라의 FOV에 따라 정해지게 된다. 우선



(a) (b)

[Fig. 2] (a) The camera is placed in the spherical coordinate system around the object and objects are placed at the origin. (b) The camera is located in the origin of Cartesian coordinate and objects are placed in the z-axis direction of the camera.

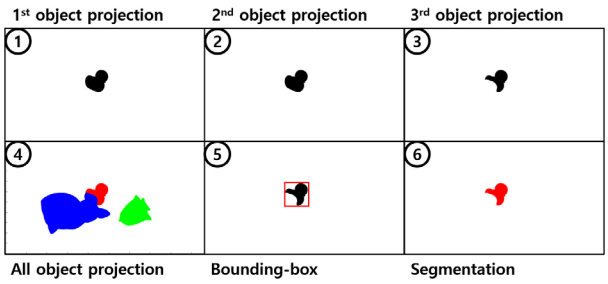
적으로, Z의 값이 정해지면 식 (1)를 통해 X와 Y의 값은 식 (2)과 같은 관계를 가지게 된다. 식 (2)에서 x와 y는 카메라의 해상도에 의해 범위가 정해져 있는 값이므로 X와 Y는 식 (3)와 같은 범위 내에서 무작위로 생성된다.

$$\mathrm{X}=\frac{x-480}{1120}\mathrm{Z},\quad \mathrm{Y}=\frac{\mathrm{y}-270}{1120}\mathrm{Z} \tag{2}$$

$$-\frac{480}{1120}Z \le X \le \frac{480}{1120}Z$$
$$-\frac{270}{1120}Z \le Y \le \frac{270}{1120}Z \tag{3}$$

3.1.4 물체의 바운딩 박스와 Segmentation Annotation

본 절에서는 이미지에 필요한 Annotation을 추출하기 위한 과정을 구체적으로 설명한다. 본 연구에서는 N개의 물체를 Virtual Camera의 이미지 평면으로 프로젝션 시키는 과정에서 물체의 색상을 조작하여 물체의 Annotation을 추출한다. 구체적으로, 카메라와의 거리에 해당하는 값인 Z가 큰 물체부터 (카메라로부터 멀리 있는 순으로) Annotation을 추출한다. 이때에 n번째로 Z값이 큰 물체의 Annotation을 추출하기 위해, n번째로 Z 값이 큰 물체를 제외하고 물체의 색상을 하얀색으로

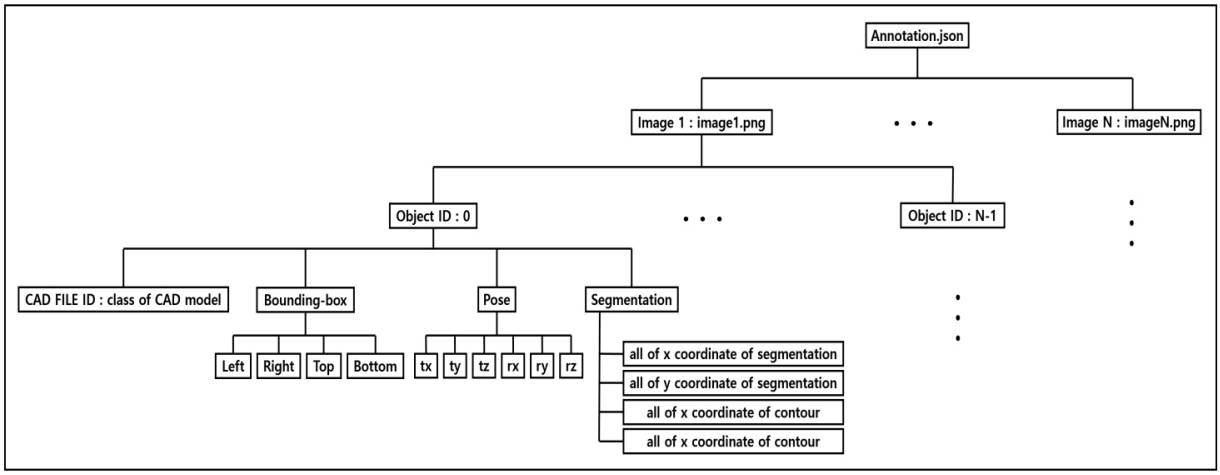


[Fig. 3] If there are three objects, this is the process of obtaining the annotation of object with the smallest Z. The first row is the process in which the object is occluded due to occlusion during the projection process. Second row is the result of extracting bounding-box and segmentation annotation of the object.

지정하여 이미지 평면에 투영한다. 예를 들어, [Fig. 3]의 4와 같이 3개의 물체가 있는 상황이라고 했을 때, 카메라와의 거리에 해당하는 Z는 Red Object > Green Object > Blue Object 순이다. Z값이 가장 큰 Red Object부터 Virtual Camera에 의해 이미지 평면으로 프로젝션되고 마지막으로 blue object가 프로젝션된다. 이때, Red Object의 Annotation을 추출하기 위해, Red Object를 제외한 나머지 물체들의 색상을 하얀색으로 하여 배경색과 같아지게 한다. 이렇게 Z값이 가장 큰 Red Object를 프로젝션 한 결과를 [Fig. 3]의 1에 표시하였다. 이후 두 번째 단계로 Green Object의 색상을 하얀색으로 하여 프로젝션하게 되면, Red Object와 Green Object 사이에 겹쳐지는 부분이 없기 때문에 [Fig. 3]의 2와 같은 결과를 얻을 수 있다. 마지막으로 Blue Object의 색상을 하얀색으로 하여 프로젝션하면, [Fig. 3]의 3과 같은 Occlusion이 생긴 물체의 형태를 얻을 수 있다. 이와 같은 방식으로 얻어진 이미지들을 이용하여 각각의 물체에 대한 Annotation을 추출하게 된다. Segmentation Annotation을 추출하기 위해, [Fig. 3]의 3과 같은 프로젝션 결과 이미지를 Gray Scale로 변환한다. 그 후, 단일 채널 이미지 픽셀들의 간단한 크기 비교를 통해 물체의 Segmentation Annotation을 얻을 수 있다. Bounding Box Annotation을 얻어내기 위해 가우시안 필터를 적용해 물체 형태의 가장자리의 모호함을 제거한다. 그 후, Canny Edge Detection을 수행하여 물체의 Contour의 좌표를 알아낼 수 있다. 이 Contour의 x, y 좌표의 최댓값과 최솟값을 이용하여 Bounding Box를 추출해 낸다. 이러한 Annotation 추출 과정은 모든 물체에 반복하여 수행한다. Annotation을 추출하는 전체 과정은 [Algorithm 1]에 제시하였다.

[Algorithm 1] Extraction of the multiple object annotation

Data : N sets of 3D CAD model and sampled object pose
Result : bounding box, contour, and segmentation of multiple object
Sort objects in order to Z;
Pose the camera at the origin;
Pose the object to fit the sampled pose;
For i=1 **to** N **do**
  For j=1 **to** N **do**
    If i **is** j
      Set the color of object[j] to black;
    Else
      Set the color of object[j] to white;
    End
    Project object to the image[j] plane;
  Convert projected image(I) to gray scale(I_g);
  For all pixel of gray scale(I_g) **do**
    If I_g(x, y) **is not** 255
      Segmentation = I_g(x, y);
  Gaussian blur(I_b) to gray scale(I_g);
  Detect edge(I_e) of gaussian blur(I_b);
  Contour = I_e;
  BB_left = min(x coordinates of I_e);
  BB_right = max(x coordinates of I_e);
  BB_right = max(x coordinates of I_e);
  BB_top = max(y coordinates of I_e);



[Fig. 4] Hierarchy of the annotation file : Initially, we divided by each image. Next, an object hierarchy is created based on the number of objects in the image. The object hierarchy contains the annotation information of the object. Object annotation contains information about the CAD model class, bounding box, pose, and segmentation.

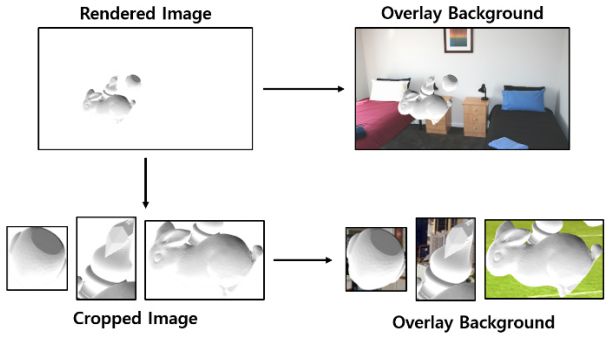
3.1.5 Annotation 저장 방식

Annotation은 json타입으로 저장하며, 각각의 이미지별로 분류한다. Annotation은 한 이미지에 나타나는 물체의 개수와 각각의 물체에 대한 정보들을 포함한다. Object Classification, Object Detection, Instance Segmentation, Object Pose Estimation을 수행할 수 있는 Annotation들을 모두 포함한다. 세부 사항은 [Fig. 4]에 표시해 두었다.

4. 이미지 합성

4.1 Rendering Image

본 연구에서는 이미지 렌더링을 위해 Blender[13]를 사용하였다. 지금부터의 과정은 위에서 얻은 Annotation들을 이용하여 물체를 정렬하고 데이터 셋에 사용될 이미지를 만드는 과정이다. 전체 과정은 [Fig. 5]에 소개되어 있다.



[Fig. 5] The process of synthesizing images : synthetic images with object mask and object class annotations are utilized for the training set of the object detection. For the pose estimation, synthetic images are cropped using truncation annotation. Then, the cropped images and pose annotations were used for the training set of the pose estimation.

본 연구에서는 다양한 광학적 환경을 만들기 위해 6~16개의 점광원(Point Light)을 추가한다. 각 점광원의 포즈는 좌표계의 원점에 위치한 카메라를 중심으로 균일하게 샘플링한다. 또한 점광원의 에너지 값은 평균과 표준편차 모두 2가 되게 샘플링한다.

Blender[13]를 통해 3D CAD 모델을 렌더링하면 투명 배경의 이미지를 얻는다 ([Fig. 5]의 Rendered Image). 이 이미지는 4.2와 4.3의 과정을 통해 연구의 목적에 맞는 이미지로 변경된다.

4.2 Cropping Image

이 과정은 Pose Estimation에 적합한 데이터 셋을 만들기 위해 필요하다. 4.1에서 얻은 이미지는 Annotation에 포함된 Bounding Box를 관심 영역으로 간주하고 이미지를 잘라낸다 ([Fig. 5]의 Cropped Image). 이를 통해 Pose Estimator는 다른 것들은 제외하고 오로지 추정해야 하는 물체에만 집중할 수 있다.

4.3 Overlay Background

4.1과 4.2에서 얻은 두 가지 이미지는 [Fig. 5]의 Rendered Image와 Cropped Image와 같이 배경을 가지고 있지 않다. 본 절에서는 [Fig. 5]의 Overlay Background와 같이 이미지에 배경을 씌운다. 배경으로 사용할 이미지는 SUN2012 PASCAL[14]에서 Scene 카테고리에 있는 이미지를 사용하였다. 트레이닝 시에 딥 뉴럴 네트워크가 지나치게 비현실적 이미지에 오버피팅되는 것을 방지하기 위해 다양한 배경 이미지를 사용하였다.

4.4 수중 환경을 반영한 이미지 합성

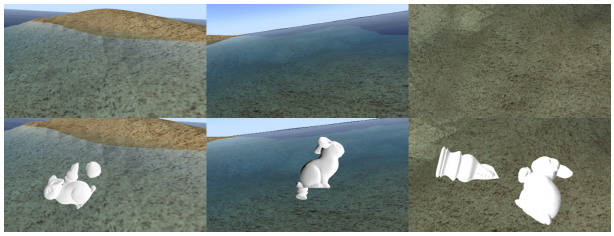
4.3까지의 과정은 일반적인 환경에서의 데이터 셋을 만드는 과정이다. 수중 환경에서의 데이터 셋을 만들기 위해 4.3에서 사용한 SUN2012 PASCAL Scene 카테고리의 배경 이미지 대신 수중 환경의 배경 이미지를 사용한다. 수중 환경 이미지는 Jaume-I Castelln 대학의 IRSLab에서 만들어진 수중 시뮬레이터인 UWSim[15]를 통해 얻을 수 있다. [Fig. 6]은 UWSim을 통해 얻은 배경 이미지 중 일부와 합성 이미지를 나타낸다.

4.5 Translating synthetic image to haze image

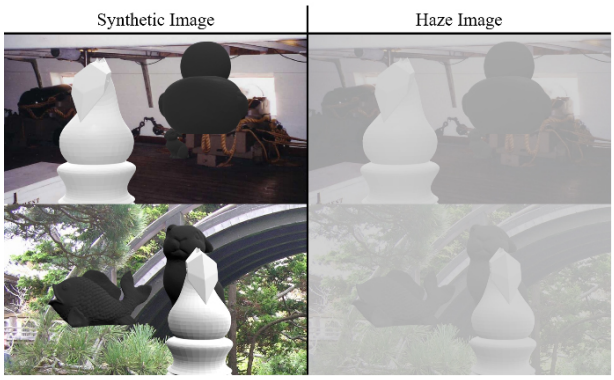
실제 수중 환경에서는 물 속에 다양한 불순물이 존재하여, 4.4에서 생성된 합성 이미지와 같은 깨끗한 형태의 이미지가 아닌 흐릿한 형태의 이미지를 얻는다. 이러한 수중 환경을 반영하기 위해 본 연구에서는 Haze Image Model[16]을 이용하여 전체적으로 흐릿해진 이미지를 생성한다.

$$I(x) = J(x)e^{-\beta} + A(1-e^{-\beta}) \quad (4)$$

식 (4)에서 I(x)는 Haze Image, J(x)는 Haze-Free Image를 의미하고, A는 Global Ambient Light, β는 Attenuation Coefficient로서 Haze의 강도를 조절하는 파라미터이다. 본 연구에서는 β를 1.5로 설정하여 [Fig. 7]과 같은 Haze Image를 얻었다.



[Fig. 6] The example of the underwater environment. The first row shows underwater background images captured in UWSim. The second row is the result of synthesizing images.



[Fig. 7] The result of translating synthetic image to haze image. The left column shows two synthetic images. The right column presents two haze images translated from synthetic images.

5. Object Detection

본 연구에서 제안한 데이터 셋이 CNN을 기반으로 한 Object Detection에 적합한지를 MASK R-CNN을 통해 테스트하였다. 또한, 4.5절에서 제안한 Haze Model을 적용한 데이터 셋이 가지는 수중에서의 효과를 검증하기 위해 2가지의 실험(Non Haze Dataset, Haze Dataset)을 진행하였다.

5.1 청수로 채워진 수조에서의 실험(Non Haze Dataset)

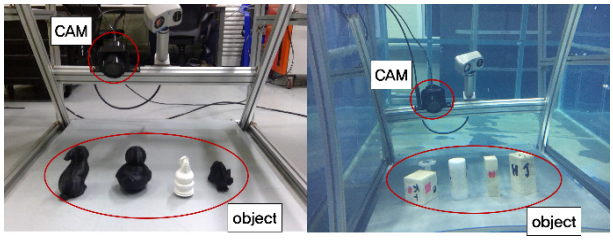
5.1.1 Experiment Setting

본 실험에 사용된 데이터 셋은 [Fig. 1]에 있는 4개의 3D CAD 모델과 UWSim의 수중 환경을 적용하여 구성하였다. 4개의 CAD 모델은 3D 프린터를 이용하여 높이 15 cm의 크기로 출력하였다. 수중 환경에서의 실험을 위해, [Fig. 8]과 같이 3D 프린터로 출력한 4개의 물체는 수조 안에 위치시키고, 수중 카메라로 그 물체들을 촬영하였다.

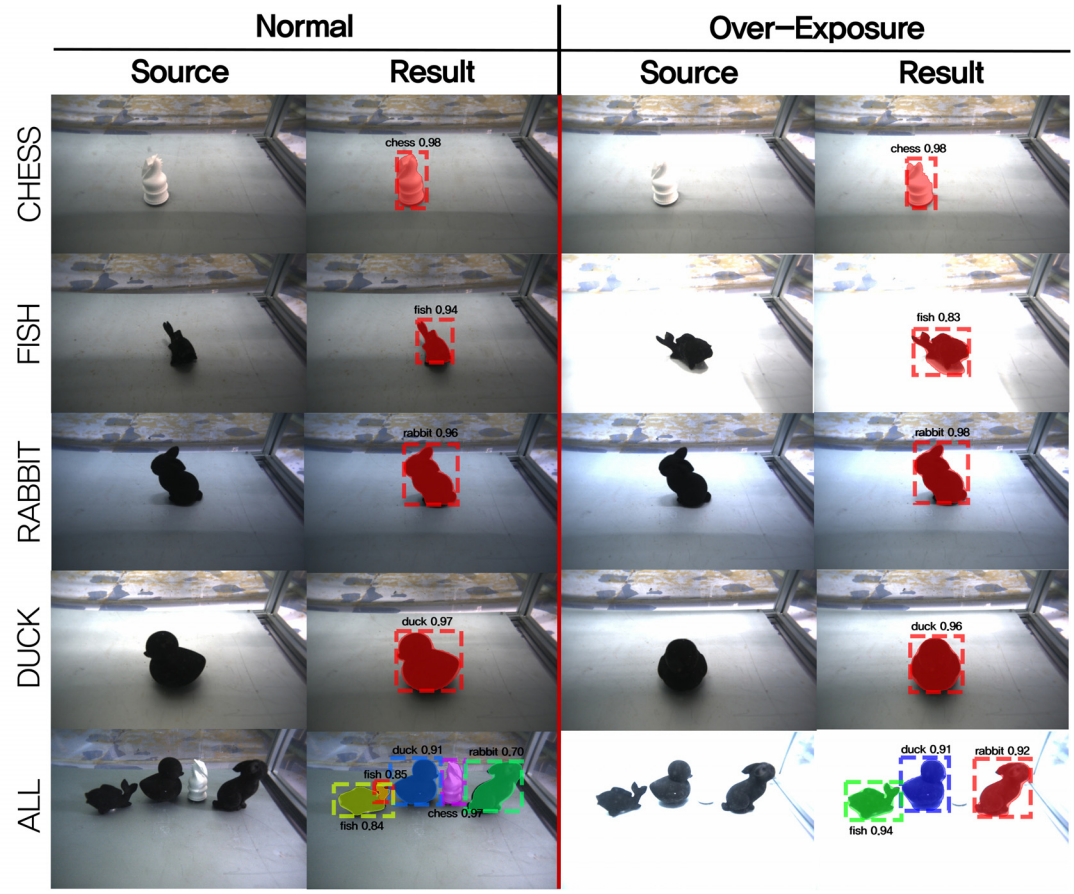
MASK R-CNN의 트레이닝을 위해 Haze Model이 적용되지 않은 총 5000장의 합성 이미지(1개의 물체를 포함한 이미지 2000장, 2개의 물체를 포함한 이미지 1500장, 3개의 물체를 포함한 이미지 1500)를 생성하였다. 각각의 물체들의 포즈는 랜덤 샘플링을 통해 생성되었다. 테스트를 위해 각각의 물체(DUCK, RABBIT, FISH, and RABBIT)는 각각 3번의 촬영을 하였으며, 촬영할 때마다 물체의 포즈를 바꿔 주었다. 또한 4개의 물체를 함께 촬영한 이미지도 테스트 셋으로 사용하였다([Fig. 9]).

5.1.2 Result of Object Detection

본 연구에서 제안하는 데이터 셋이 Underwater Object Detection에 적합한지에 대한 여부를 평가한다. 이를 위해 합성 이미지로 트레이닝 된 MASK R-CNN을 통해 얻은 Object Detection 결과는 mAP (mean Average Precision)와 Mask



[Fig. 8] Experiment setup. a camera and four objects are prepared in-air (left), and then placed in-water (right)



[Fig. 9] Object detection results. The results of the four objects and the all four object are shown. The first two columns are the source image and the detection results in the normal situation. The third and fourth columns are the detection results in the over-exposure situation. In the results column, the class name and detection score are represented, and the colored region represents detection mask. The bounding box of object is depicted by dash line.

Overlay를 사용하여 정확도를 평가하였다. mAP는 Pascal Visual Object Class (VOC) Challenge에 사용된 객체 검출기의 정확도를 측정하기 위한 기준으로 사용되었다. mAP에 사용된 Intersection over Unit (IoU)은 0.5로 설정한다. 또한, Ground Truth Mask와 예측된 Mask의 겹치는 비율(Mask Overlay)은 Segmentation의 정확도를 측정하는 데에 사용되었다.

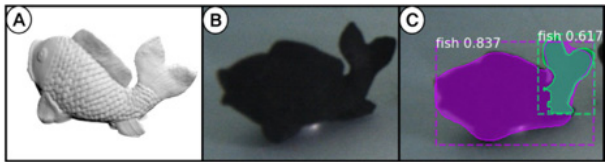
[Table 1]에서 볼 수 있듯이, 본 연구에서 제안하는 데이터셋으로 트레이닝된 Object Detector는 mAP와 Mask Overlay가 FISH 모델을 제외하면 0.8을 넘는 높은 정확도를 가지는 것을 볼 수 있다. Fish 3D 모델은 다른 모델보다 훨씬 디테일하게 표현되어 있어 비늘 부분까지 확인할 수 있고, 합성이미지에 FISH 모델의 디테일이 표현된다([Fig.9]의 A). 이러한 이유로,

[Table 1] Summary of object detection with evaluation metrics.

Object	Chess	Duck	Rabbit	Fish	All
mAP	0.86	0.82	0.88	0.64	0.81
Mask Overlay	0.87	0.89	0.83	0.71	0.82

MASK R-CNN은 트레이닝 과정에서 FISH 모델의 비늘 부분에서 Low-Level Keypoint를 추출한다. 그러나, 수중에 있는 카메라는 FISH 모델의 비늘 부분과 같은 디테일을 촬영해 내지 못한다 ([Fig. 10]의 B). 결론적으로, FISH 모델의 꼬리 모양과 같은 High-Level Keypoint들을 통해 FISH 모델을 인식하게 되고 [Fig. 10]의 C와 같은 잘못된 결과를 얻게 되었다.

[Fig. 9]에서 Source Column은 수중에서 촬영한 이미지의 원본이고, Result Column은 Object Detection의 결과이다. 인식



[Fig. 10] Detection problem of the fish model. (A) is the synthetic image using fish model. (B) shows a printed model photographed in underwater. (C) is the object detection result of this model. The object class detected by MASK R-CNN is presented as colored regions. Detection score is presented by number.

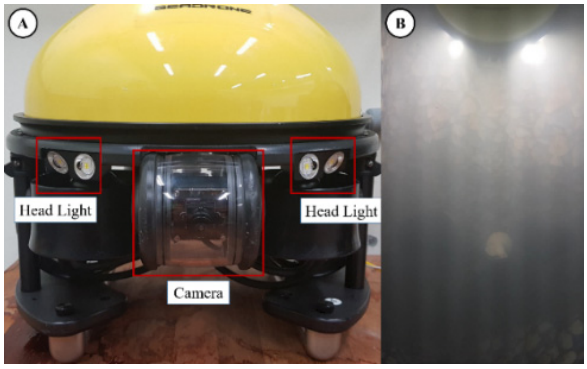
된 물체는 색상을 통해 Segmentation의 정보를 표현하고, 빨간색의 Bounding Box를 그려 주었다. 본 실험은 다양한 광학적 조건에서 실행되었으며, 2번째와 4번째 행에서 볼 수 있듯이 Over-Exposure 상황에서도 높은 정확도를 갖는 것을 확인할 수 있다.

5.2 실제 수중 환경을 반영한 수조에서의 실험(Haze Dataset)

5.2.1 Experiment Setting

본 연구에서 제안하는 데이터 셋의 유효성을 검증하기 위해 청수로 채워진 수조가 아닌 실제 수중 환경을 최대한 반영한 환경에서의 실험이 필요하다. 이에 따라 본 실험에서는 [Fig. 11]과 같이 환경을 구성하였다. 우선적으로 외부로부터의 빛을 차단하기 위해 수조에 암막 커튼을 설치하였다. 또한 수중 환경의 배경 효과를 위해 자갈 모양을 출력한 천을 수조 안에 설치하였다. 최종적으로, 수조에 담긴 물에 불순물을 첨가하여 물의 농도를 탁하게 만들어 줌으로써 실제 수중 환경과 유사하게 만들어 주었다.

[Fig. 11] Experiment Setup. The curtain is installed to block the light from the outside. The gravel background is built in the water tank.



[Fig. 12] SeaDrone. (A) Sea Drone has four head lights and one camera. (B) The head lights are turned on to gain visibility.

실제 수중에서 수중 로봇이 이미지를 촬영하는 것과 같은 환경을 반영하기 위해 SeaDrone[17]을 활용하였다. 본 실험에서는 외부의 빛을 모두 차단한 채로 진행을 하기 때문에 SeaDrone의 Head Light를 사용하여 시야를 확보하고 SeaDrone에 부착된 카메라를 이용하여 이미지를 촬영하였다([Fig. 12]).

MASK R-CNN의 트레이닝을 위해 생성한 합성 이미지는 Haze Model이 적용되었으며, 5.1.1절에서 수행한 실험과 같은 개수로 생성하였다.

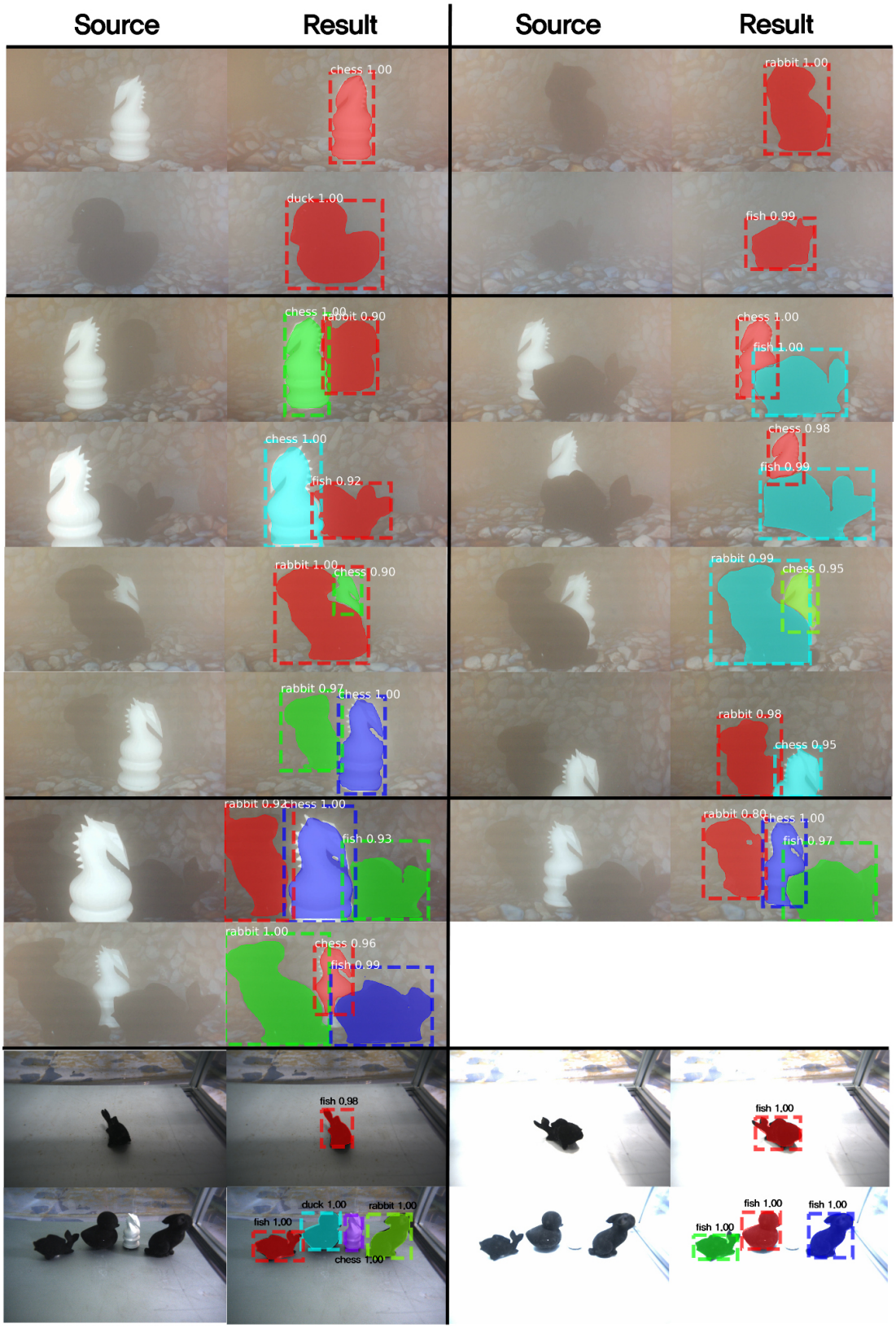
5.2.2 Result of Object Detection

본 연구에서는 소형 ROV, SeaDrone을 활용하여, 실제 AUV가 수중에서 물체를 인식할 때에 본 연구가 제안하는 데이터 셋으로 트레이닝된 시스템에 대한 유효성을 검증하기 위해 실험을 진행하였다. [Fig. 13]에서 Source Column은 수중에서 촬영한 이미지의 원본이고, Result Column은 Object Detection의 결과이다. 인식된 물체는 색상을 통해 Segmentation의 정보를 표현하고, 빨간색의 Bounding-Box를 그려주었다. 본 실험은 물체들 간의 Occlusion을 고려하여 물체를 배치하여 실험하였다. 5.1.2에서 사용한 성능 평가 지표를 본 실험에서도 동일하게 이용하여 성능을 평가하였고, [Table 2]에 나타내었다. Self-Occlusion을 제외하면 Occlusion이 존재하지 않는 단일 물체일 때에 mAP와 Mask Overlay가 전반적으로 0.9를 상회하는 높은 정확도를 가지고 있으며, Occlusion이 발생한 상황에서도 청수에서의 실험(5.1절) 결과와 견줄만한 수치적 정확도를 가진다. 또한 [Fig. 13]를 통해 물체가 50% 가까이 Occlusion이 생겼음에도 불구하고 Object Detection이 정확하게 수행된 것을 확인할 수 있다.

본 실험에서 사용한 데이터 셋은 5.1.2의 데이터 셋보다 실제 수중 환경과 더 비슷한 조건을 만들기 위해 Haze Model을 사용하여 트레이닝 이미지에 불순물을 첨가하였다. 이를 통해 실제 수중 환경과 본 연구에서 제안하는 데이터 셋 사이의 Reality Gap이 줄어들어 Object Detection의 정확도가 높아진 것을 확인할 수 있다. 뿐만 아니라, Haze가 적용된 이미지가 FISH 모델의 시각적 디테일을 감소시켜 청수에서 발생했던 FISH 모델의 인식 문제도 개선된 것을 볼 수 있다([Fig. 13]).

[Table 2] Summary of object detection with evaluation metrics.

	Occlusion		Non-Occlusion	
	mAP	Mask Overlay	mAP	Mask Overlay
Chess	0.754	0.865	0.913	0.916
Duck	0.750	0.864	0.936	0.942
Rabbit	0.794	0.876	0.921	0.925
Fish	0.719	0.843	0.865	0.892
All	0.754	0.862	0.908	0.918



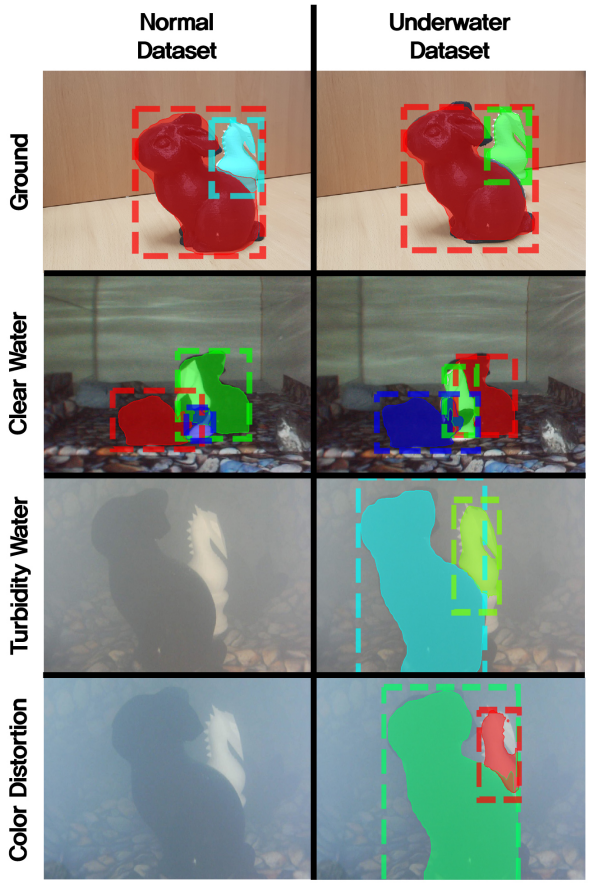
[Fig. 13] Object detection results in the watertank reflecting the actual underwater environment. The class name and detection score are represented, and the colored region represents detection mask. The bounding box of object is depicted by dash line The object detection results were displayed in color and detection scores. The result in the case of occlusion between the objects is shown from the third to the sixth row. The result in the case of clear water is shown in the last two rows.

5.3 Comparison between Normal and Underwater Dataset

본 절에서는 일반적인 환경을 위하여 생성된 데이터 셋과 본 연구에서 제안하는 수중 환경을 반영한 데이터 셋의 비교를 통해, 일반적인 환경을 위한 데이터 셋이 가지는 수중에서의 한계와 수중 환경을 반영한 데이터 셋의 필요성을 보인다.

일반적인 환경을 위한 데이터 셋은 [7]에서 제안된 방식을 사용하여 5.1.1절에서 수행한 실험과 같은 개수로 생성하였다.

본 연구에서 제안한 수중 환경 데이터 셋과 일반적인 환경을 위한 데이터 셋의 간단한 시각적 비교를 [Fig. 14]에 나타내었다. 4가지의 케이스(Ground, Clear Water, Turbidity Water, Color Distortion)를 통해 두 데이터 셋을 비교하였다. 수중 환경 데이터 셋으로 트레이닝된 모델은 모든 케이스에서 좋은 인식 결과를 보이는 것을 확인할 수 있다. 그러나, 일반적인 환경을 반영한 데이터 셋으로 트레이닝된 모델은 Clear Water의 케이스에서 겹침 현상이 발생하는 경우 물체 인식에 어려움을 겪는 것을 확인할 수 있다. 또한, Turbidity Water와 Color Distortion의 케이스에서도 물체를 인식하지 못하는 것을 볼 수 있다.



[Fig. 14] Comparison between normal and underwater dataset. The first column is the object detection result using the model trained by the normal dataset. The result of the model trained by underwater dataset is shown in the second column. Each dataset is evaluated in the four cases. The result of the object detection is shown by the object mask and bounding box.

6. 결 론

본 연구에서는 3D CAD 모델을 이용하여 물체들 간의 Occlusion을 반영한 데이터 셋을 만들고, 그 데이터 셋에 Annotation을 자동으로 추출하는 방법을 제안한다. 본 논문에서 제안하는 데이터 셋은 일반적인 환경뿐만 아니라 수중환경에서도 이용될 수 있다는 점에서 차별성이 있다. 구축된 데이터 셋을 통해 우리는 MASK R-CNN을 사용하여 Object Detection의 성능을 테스트 하였고, 실험을 통해 본 연구가 제안하는 데이터 셋이 Underwater Object Detection에 적합하다는 것을 보였다. 후속 연구로는 환경 특성에 맞는 이미지 표현을 위해 합성된 이미지의 후처리에 대한 연구가 필요하다. 본 연구에서 제안하는 학습방법을 추후 SLAM 및 수중 센싱 연구[18, 19]에 활용할 예정이다.

References

[1] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, DOI: 10.1109/CVPR.2014.81.

[2] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154-171, 2013.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, DOI: 10.1109/CVPR.2016.91.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, pp. 2980-2988, 2017.

[5] R. Girshick, “Fast R-CNN,” *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, pp. 1440-1448, 2015.

[6] X. Peng, B. SUN, K. Ali, and K. Saenko, “Exploring Invariances in Deep Convolutional Neural Networks using Synthetic Images,” *arXiv:1805.12177v2*, 2014.

[7] H. Su, C. R. Qi, Y. Lim, and L. J. Guibas, “Render for CNN: Viewpoint Estimation in Images Using CNNs Trained with Rendered 3D Model Views,” *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, pp. 2686-2694, 2015.

[8] Trimble Inc, *3D Warehouse*, [Online], https://3dwarehouse.sketchup.com, Accessed: March 19, 2019.

[9] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan, “Driving in the Matrix: Can virtual worlds replace human-generated annotations for real world tasks?,” *2017 IEEE*

International Conference on Robotics and Automation (ICRA), Singapore, Singapore, 2017, DOI: 10.1109/ICRA.2017.7989092.

[10] H. Hattori, N. Lee, V. N. Boddeti, F. Beainy, K. M. Kitani, and T. Kanade, "Synthesizing a Scene-Specific Pedestrian Detector and Pose Estimator for Static Video Surveillance," *International Journal of Computer Vision*, vol. 126, no. 9, pp. 1027–1044, Sept., 2018.

[11] P. P. Busto and J. Gall, "Viewpoint refinement and estimation with adapted synthetic data," *Computer Vision and Image Understanding*, vol. 169, pp. 75–89, Apr., 2018.

[12] Y. Wang, X. Tan, Y. Yang, X. Liu, E. Ding, F. Zhou, and L. S. Davis, "3D Pose Estimation for Fine-Grained Object Categories," *European Conference on Computer Vision*, pp. 619-632, 2018.

[13] Stichting Blender Foundation, Blender, [Online], http://www.blen der.org, Accessed: March 19, 2019

[14] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, USA, pp. 3485-3492, 2010.

[15] M. Prats, J. Pérez, J. J. Fernández, and P. J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Portugal, pp. 2577-2582, 2012.

[16] Y. Cho and A. Kim "Channel invariant online visibility enhancement for visual SLAM in a turbid environment," *Journal of Field Robotics*, vol. 35, no. 7, pp. 1080-1100, 2018.

[17] SeaDrone Inc, *SeaDrone*, [Online], https://seadronepro.com, Accessed: March 19, 2019.

[18] Y. Lee, J. Choi, and H-T. Choi. "Underwater Robot Localization by Probability-based Object Recognition Framework Using Sonar Image," *Journal of Korea Robotics Society*, vol. 9, no. 4, pp. 232-241, Nov., 2014

[19] Y.-S. Shin, Y.-J. Lee, H-T. Choi, and A. Kim, "Bundle Adjustment and 3D Reconstruction Method for Underwater Sonar Image," *Journal of Korea Robotics Society*, vol. 11, no. 2, pp. 051-059, Jun., 2016.

전 명 환

2017 광운대학교 로봇학부(학사)

2017~2018 실감교류인체감응솔루션연구단 위촉연구원

2018~현재 KAIST 로봇공학학제전공 석사과정

관심분야: 로봇 비전, 수중 로봇

장 혜 수

2018 한국과학기술원 건설 및 환경공학과 (학사)

2018~현재 한국과학기술원 건설 및 환경공학과 석사과정

관심분야: 수중 로봇, 로봇 비전, SLAM

이 영 준

2009 충남대학교 메카트로닉스공학과(학사)

2014 충남대학교 메카트로닉스공학과(석사)

2011~현재 한국해양과학기술원 부설 선박해양플랜트연구소 기술원

관심분야: 수중로봇, 자율탐사, 소나기반물체인식

여 태 경

1998 부산수산대학교 기계공학과(공학사)

2000 부경대학교 기계공학과(공학석사)

2003 쿠마모토대학 시스템정보공학과(공학박사)

2005~현재 선박해양플랜트연구소 책임연구원

관심분야: 수중 로봇 설계 및 제어

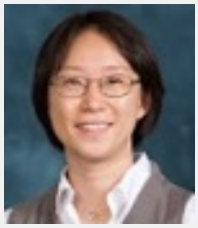
신 영 식

2013 인하대학교 전기공학과(공학사)

2015 KAIST 로봇공학학제전공(공학석사)

2015~현재 KAIST 건설 및 환경공학과 박사과정

관심분야: SLAM, 로봇 비전, 수중 로봇



김 아 영

2005 서울대학교 기계항공공학과(공학사)

2007 서울대학교 기계항공공학전공(공학석사)

2012 미시간대학교 기계공학전공(공학박사)

2014~현재 한국과학기술원 건설 및 환경공학과 조교수

관심분야: 영상기반 SLAM