

橋脚修理部材運搬用自律水上ロボットのためのターゲット認識 Target Detection for Autonomous USVs Transporting Repair Material of Bridge Piers

○根本 幹大¹, 竹入 将平¹, 李 賢品², 河内山 学², 西村 光生², 羽多野 正俊³

*Kanta Nemoto¹, Shohei Takeiri¹, Kenpin Ri², Manabu Kawachiyama², Mitsuo Nishimura², Masatoshi Hatano³

Abstract: Explorations and repairs of bridge piers in ports are accomplished by divers in general. However, environments around the target areas are dangerous because of waves. Therefore, our final goal of the research is to develop an autonomous USV instead the divers. In this presentation, we show an object detection method using the SSD and experimental results.

1. 緒言

日本は海洋国家であり、全国に港湾が存在するが、それらの多くは築 50 年を超え、長年海中に没した部分は海水によって腐食され、修理が急務となっている。現在の修理の多くは、橋脚全体の取り替えではなく、クラック等の入った部分に金属板をパッチとして取り付ける手法が多く用いられており、この修理部品を人力で海上を運搬している。

本研究では、この橋脚修理部品運搬用の水上ロボットである USV(Unmanned Surface Vehicle)の開発を目的とする。開発する USV は自律型で、ターゲット(目的地の目印)の位置検出を行い、その方向に自動誘導によって推進する。その際、自然画像の背景からターゲットを正しく検出する問題や、水面からの光の乱反射の影響によりカメラ映像がフレアを起こしやすい環境下においてターゲットを検出する問題があり、これに対し、AI(Artificial Intelligence)を用いた手法を考える。本報告では、制御システムと物体検出手法について述べる。

2. 物体検出手法および環境構築

ターゲットの物体検出を行う手法として SSD(Single Shot MultiBox Detector)の Keras/TensorFlow で実装されたソースを利用した。SSD は、畳み込みニューラルネットワークを用いた物体検出アルゴリズムである。

開発環境は以下のようなものである。

- Ubuntu 14.04 LTS:Debian をベースとした OS で, Linux ディストリビューションである。

- python 3.6.2 : 汎用のプログラミング言語である。コードがシンプルで扱いやすく設計されており, C 言語等に比べて, 様々なプログラムを分かりやすく, 少ないコード行数で書ける。

- Anaconda3-5.2.0-Linux-x86_64 : python 本体と, python でよく利用されるライブラリをセットにした python パッケージである。

- opencv-python 3.4.1.15 : オープンソースのコンピュータ・ビジョン・ライブラリである。コンピュータで画像や動画を処理するのに必要な, 様々な機能が実装されている。

- tensorflow 1.0.0 : Google が開発しオープンソースで公開している, 機械学習に用いるためのソフトウェアライブラリである。[1]

- keras 2.0.2 : python で書かれた, tensorflow または CNTK,theano 上で実行可能な高水準のニューラルネットワークライブラリである。

3. 製作した USV

Fig. 1 に、製作した USV を示す。ロボットボディは簡易で強度および浮力が大きく取れるペットボトルとアルミ材を用いて製作し、両側に推進用のモータ・プロペラを用いたスラストが取り付けられている。またボディ上部には USB カメラが取り付けられており、自律走行用の画像取得に用いられる。さらに、修理部材を模したアルミ板がロープによりロボット下部に取り付けられている。

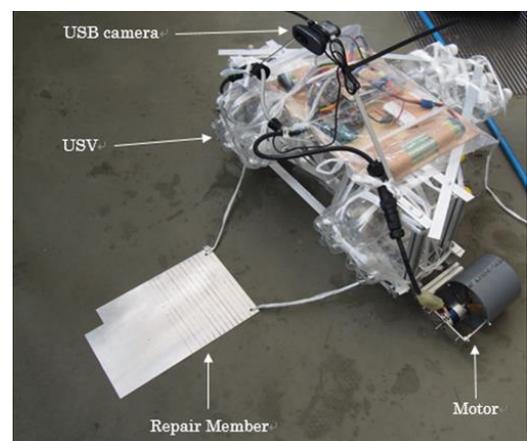


Fig.1 Constructed USV

ロボット本体には、制御用の PC と Arduino が設置され、これらの駆動用バッテリーおよびスラストの駆動バッテリーも設置されている。これらの機材はビニール製の防水袋に入れられている。また、水上走行時に後傾にならないように、ロボット本体の後ろにもう一つの浮体を設置している。この水上ロボットを走行させ、修理部材を目的地まで運ぶ。

4. 制御システム

Fig. 2 に構築した制御システムを示す。PC に Arduino が接続されている。操縦信号はシリアル通信にて Arduino に送られ、そこから PWM 通信としてモータに送られ、スラストが動作する。また Arduino には GPS が接続され、その位置情報が PC に送られる。この PC には USB カメラが接続され、リアルタイムで物体検出が行われる。

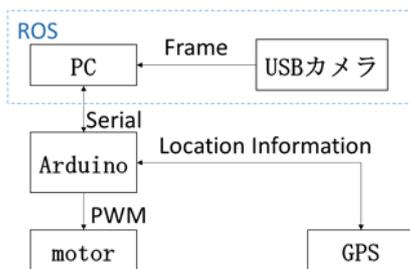


Fig.2 Control System

5. ターゲット検出実験

SSD を用いて物体検出が可能かどうか、および検出可能な距離を求める実験を行った。実験用プールの対岸の距離(約 8m)から物体検出可能な目標距離を約 7~9m とした。USV を海で稼働させたことを想定した場合、一番目立ち海面に浮く最適なターゲットとして赤いボールが考えられる。よって、これを物体検出するターゲットとして決定した。

ターゲットが写った画像ファイル(125×94)を 262 枚用意した。Processing3.3 のツールを用いて、ターゲットの位置情報を xml データ(アノテーションデータ)に変換する。学習に必要なデータとして、学習済みの重みやバイアスを保存した weights_SSD300.hdf5 を入手する。The PASCAL Visual Object Classes Homepage に準備(公開)されている学習用データ VOC2007 を利用し、このファイル内に既存する jpg ファイルと xml ファイルを独自に準備したものと入れ替えることにより、計算済みのデフォルトボックスの位置を保存した pkl ファイルを作成できる。この pkl ファイルを

weights_SSD300.hdf5 に学習させることにより、独自の学習モデルを作成できる。

Fig.3 に実験環境を示す。物体検出している様子を Fig. 4 に示す。カメラを設置し、ターゲットを撮影する。ターゲットを所持した実験者がカメラから徐々に離れていき、認識できているかどうか調べた。場所は、日本大学理工学部船橋キャンパス 6 号館 1 階廊下、晴天下の屋内で実験を行った。

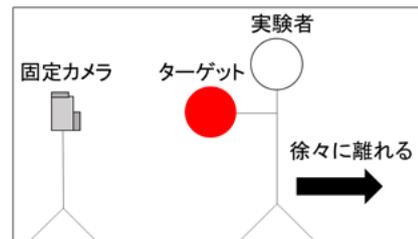


Fig.3 Experiment Environment



Fig.4 State at Object Detection

実験結果として、カメラからターゲットまでの距離が約 15m 以内であれば物体検出が可能であった。しかし、検出されないタイミングがあり、ターゲットに当たる日光の角度や影等の影響により形状欠損を起したことが原因だと考えられる。この問題に対しては、様々な光環境におけるターゲットの画像データを用意し、学習することにより性能向上が期待できる。

6. 結言

本報告では、橋脚修理部材用運搬用水上ロボットのための基礎研究として、USV の制御システムの概要と物体検出の結果を示した。

7. 参考文献

[1] 新村拓也(2018)

『TensorFlow ではじめる DeepLearning 実装入門』
株式会社インプレス