

ROV と AI による海中生物モニタリングシステムの研究開発

Research and Development of Marine Life Monitoring System using ROV and AI

研究代表者 沖縄工業高等専門学校, 情報通信システム工学科 教授 中平勝也,

Katsuya Nakahira

Traditionally, marine life surveys have been conducted by survey divers. This paper proposes a system that uses ICT technology to survey marine life automatically and remotely. The captured undersea images are converted into clearer undersea images using super-resolution and image correction technologies. Fish and corals are detected and monitored using neural networks. In the underwater drone system. An underwater drone captures underwater images and transmits them to a server from a transmitter installed on an ocean buoy to detect fish and corals. Finally, fish and corals are detected and monitored from those images by a neural network. Our proposed marine life survey system enables low-cost, wide-area, long-duration, and highly accurate surveys.

要旨

従来、海洋生物の調査は調査ダイバーによって行われてきた。本稿では、ICT 技術を活用し、海洋生物の調査を自動的かつ遠隔で行うシステムを提案する。撮影された海中画像は、超解像技術や画像補正技術により、より鮮明な海中画像に変換される。ニューラルネットワークを用いて魚や珊瑚を検出し、監視する。水中ドローンシステムでは水中ドローンが海中映像を撮影し、海洋ブイに設置した送信機からサーバーに送信することで、魚やサンゴを検出します。最後に、その画像からニューラルネットワークで魚やサンゴを検出し、監視する。提案する海洋生物調査システムは、低コスト、広域、長時間、高精度の調査を可能にする。

1. まえがき

地球温暖化や海洋プラスチックごみが海中の生態系へ与える悪影響が問題となっている。環境省は定期的にサンゴが海底を覆う割合を調査しており、サンゴが減少している地域があることが報告されている^[1]。サンゴの育成状況は海洋環境のバロメーターである。このような状況から、海中生物の調査を詳細かつ継続的に行い、必要な改善があればその都度対応することが大変重要である。

水深 10m~20m に生息する海中生物の調査は、ベルトトランセクト法とコドラート法が多く用いられている^[2]。ベルトトランセクト法は、ダイバーと船を

ベルトで結びつけ、ダイバーを牽引し、目視できる海中生物を水中ノートに記録していく。この方法は、手間と調査費用が莫大となり、1 日で数十万円以上となる。コドラート法は、海中に 1m 程度の大きさの方形区画であるコドラートを設定し、その中に存在する海中生物を調査する。この方法は、広い海域の調査が困難である。

従来の海洋調査の問題点を解決するため我々は、ROV と畳み込みニューラルネットワークを組み合わせることで、人手に頼らずに海中生物の分布を地図上にマッピングする海中生物調査システムの実現を目指している^[3]。

2. システム構成

提案する海中生物調査システムを図 1 に示す。ROV は水上ブイと通信ケーブルで接続し、水上ブイと地上のセンター局は無線回線で接続する。水上ブイのカメラから得た画像と、ROV のカメラから得た画像はディープラーニングサーバーに送信する。ディープラーニングサーバーは、ROV のカメラ画像から海中生物の情報として魚やサンゴの種類・数・密度を推定する。さらに、ディープラーニングサーバーは、水上ブイのカメラ画像から ROV の位置の特定とあらかじめ設定した目標ポイントへの制御情報の算出を行い、ROV に送信する。ROV は制御情報に従って動作する。このように、海中生物調査システムは、ROV の推定した位置を利用して、調査海域内で ROV を自

動的に航行させる。その上で、ROV のカメラ画像で認識した海中生物の情報を蓄積する。

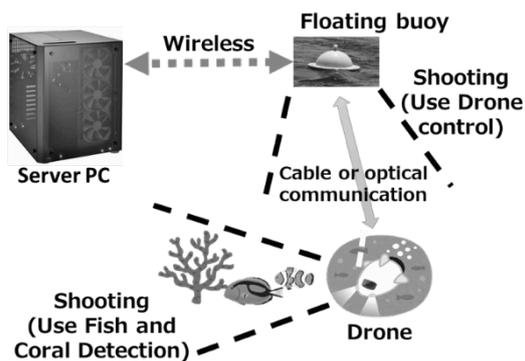


図1 ROV の航行ルート自動制御システム

3. 海洋生物の画像検出 AI

海洋生物調査では、海中に生息する海洋生物の種類・個体数・分布などの調査を行う。そこで我々は、撮影した海中の画像からニューラルネットワークで魚やサンゴの種類や数や分布を検出する AI の開発を行った。

3.1. 魚の画像検出

機械学習を用いた魚の種類認識が行われている^[4]。しかし、その精度は低く、かつ種類数が少ない。そこで、魚の画像検出は、近年発達が目覚ましいニューラルネットワークを用いた物体検出モデル YOLOv3^[5] で開発を行った。豊富な種類の魚の調査を行うため、インターネットからデータクロウリングおよびシュノーケリングで撮影した、79 種類 9525 枚の魚の画像から魚のデータセットを作成し、YOLOv3 での機械学習を行った。

図 2 は、79 種類の魚の機械学習の回数と検出精度の関係である。機械学習は 3500 回で収束し、魚の種類別に数%から 100%の検出精度となった。また、検出精度の平均値である mAP は約 65%となった。この検出精度は、30fps の動画の場合だと 1 秒間に魚が検出されるフレーム数は $30 \times 0.65 = \text{約 } 20$ フレームであり、海洋生物の調査には十分な精度ある。

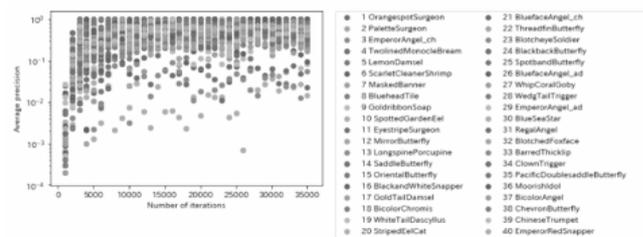


図 2 魚の画像検出の機械学習

3.2. サンゴの画像検出

サンゴは複雑な形状で広範囲に生息している。従って、前述の YOLOv3 でのサンゴの分類および領域を検出することは困難である。そこで、検出対象を領域ごとに分類するセグメンテーション型 CNN である DeeplabV3Plus を用いてサンゴの種類と領域を検出する AI の開発を行った。

機械学習において必要なサンゴの画像はインターネット上には少なかつたため、3D ゲームエンジンである Unity を活用し、サンゴが撮影された海中画像を模擬的に収集した。ここでは、3種類のサンゴ約 500 枚のデータセットを収集し、サンゴの種類と領域を分類するアノテーション作業を行った。このアノテーション情報とサンゴの画像用いて DeeplabV3Plus での機械学習を行った。

検出結果の例を図 3 に示す。AI の出力結果から、ユビサンゴ (緑)、カワラサンゴ (赤)、ミドリイシサンゴ (黄色) の検出が正確にできていることが確認できた。下側の画像は、実際に撮影したユビサンゴと画像検出結果である。実際のサンゴも検出できることも確認できた。正解の領域と検出結果の領域がどれだけか重なっているかを表す IoU (Intersection over Union) は平均 73%となった。

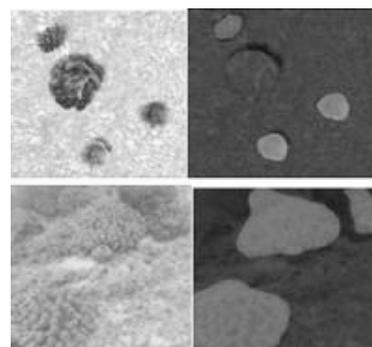


図 3 サンゴの検出結果

3.3. 超解像技術による画質の改善

海中画像には、魚やサンゴの海洋生物が撮影されているが、小さく撮影されている場合があり、その場合は画像を拡大する必要がある。このとき、拡大後の画像がモザイク画像のようになり、画像検出AIの検出精度が低下すると予測される。そこで、超解像技術を用いて海中画像の解像度を高めることで小さく撮影された海洋生物を拡大した場合でも画質が低下しないようにする。

例えば、レンズの水平画角が 40 度、垂直画角が 35 度のカメラで水深 5m の海中を撮影すると水平方向と垂直方向の撮影範囲は

$$(\tan 20 \times 5) \times 2 = 3.6\text{m} \quad (1)$$

$$(\tan 17.5 \times 5) \times 2 = 3.1\text{m} \quad (2)$$

となり 3.1m×3.6m の範囲を撮影した海中画像が得られる。海中画像の解像度を 720pix×1080pix だとすると、画像 1 ピクセルの撮影範囲は

$$\text{縦} : 310\text{cm} / 720 = 0.43\text{cm} \quad (3)$$

$$\text{横} : 360\text{cm} / 1080 = 0.3\text{cm} \quad (4)$$

となる。このとき、15cm×15cm の範囲内に魚がいると仮定すると式 (3) と式 (4) の結果を用いると、1 匹の魚は、次式のように 35pix×50pix の画像で表現されることになる。

$$\text{縦} : 15\text{cm} / 0.43\text{cm} \approx 35\text{pix} \quad (5)$$

$$\text{横} : 15\text{cm} / 0.3\text{cm} \approx 50\text{pix} \quad (6)$$

図 4 の左は、35pix×50pix で表された魚のイメージである。このような画像からは、魚の画像検出が困難であることが容易に推測できる。

そこで、超解像技術を利用し、より高画質の画像に変換する。例えば、2 倍の解像度の 70pix×100pix 画像に変換した場合、右図のようになり、魚の画像検出が可能になる。

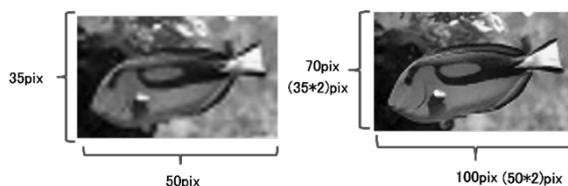


図 4 解像度の低い魚の画像(左)と超解像を利用した魚の画像(右)

3.4. 画像補正技術による画質の改善

本システムでは、カメラから調査対象の魚やサンゴまでの距離は、5~10m と想定している。海中画像は、海水の透明度が低いとぼやけて撮影される場合が多い。撮影対象までの距離が長くなると透明度は、さらに低下してく。従って、本システムでは海中画像をそのまま画像検出 AI に入力すると、海中画像に写る魚とサンゴがぼやけてしまい、魚とサンゴの検出精度が低くなる可能性がある。

そこで、画像補正技術により、海中の画像の画質の補正を行う。図 5 の左は透明度が低いときの海中画像である。図 6 は、その画像の RGB ヒストグラムである。RGB ヒストグラムの横軸は色の明るさを 256 段階で示す輝度値であり、縦軸は輝度値の出現頻度を表している。RGB ヒストグラムから、透明度が低い海中画像は、赤色の輝度値が低いことが分かる。

次に、図 5 の右は、同図の左の赤色の輝度を上げる画像補正を行った結果である。このように、赤色の輝度を上げることで、仮想的に透明度が高い海中画像に変換できる。この処理した後の海中画像を画像検出 AI に入力することで、魚とサンゴの検出精度の低下を防ぐことができ 1。



図 5 画像補正技術で目指す海中画像の例

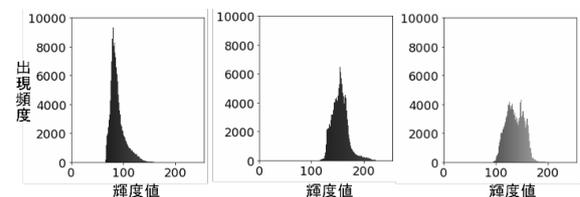


図 6 前図(左)の RGB の分布

4. ROV の航行ルート自動制御方式

近年、ROV を海中調査に利用するケースが増えてきている^[6]。しかし、海中を航行している ROV は、陸上からは直接目視することができない。そこで、ROV の操作は、ROV に装備したカメラからの水中画像を頼りに操縦者の勘で行われている。この時、操縦者のミスにより海中で ROV が岩礁に衝突し、サンゴを

傷ついたり、ROV が故障するなどのトラブルの原因となっている。

このような問題点を解決するため、ROV の位置を陸上から把握し、設定したルートに沿って ROV を自動的に航行させる方式を開発する。

4.1. 従来の ROV 位置推定

ROV の位置検出には、音響測位を用いた方法が多く用いられている。水上のハイドロフォンから ROV に取り付けられたトランスポンダに音波を送信する。音波を受信したトランスポンダは応答信号を返す。水上では送信信号と受信信号の時間差と位相差から ROV の位置推定を行う。しかし、ROV のモーターから発生する雑音によって音響測位に悪影響を及ぼし、位置推定の精度が一時的に下がる問題点がある^[7]。

4.2. 提案する航行ルート自動制御方式

図7に示すように、画像から ROV の位置を推定し、その情報をもとに ROV が航行するルートの自動制御を行う方式を提案する。ROV のモーターの雑音による誤動作を回避できるだけでなく、ROV にトランスポンダを搭載する必要がない利点がある。

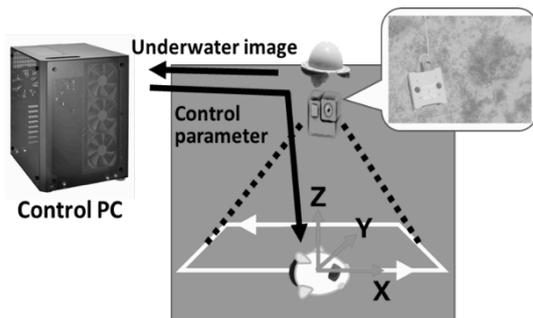


図6 ROV 航行自動制御

4.2.1. 画像による ROV の位置推定

まず、水上ブイに水中カメラを取り付け、水上ブイの下で動作している ROV を撮影する。次に、撮影した画像をディープラーニングサーバーに送信する。次に、制御 PC では取得した画像に畳み込みニューラルネットワーク (CNN) による物体検出を適用して、ROV が画像内のどこに写っているかを検出する。CNN による検出結果は、ROV を囲う矩形で示される。そこで、矩形の中心座標を ROV の X, Y 座標とする。また、ROV が水面に近づくほど画像内では大きく写る

ため、矩形の面積から Z 座標を特定する。

4.2.2. 画像による ROV の方向制御

図7に示すように、制御 PC は、取得した画像に CNN による物体検出を適用し、ROV の方向ベクトル (ROV の先端がどちらを向いているかを示すベクトル) と目標ベクトル (ROV から目標地点へのベクトル) を求める。次に、方向ベクトルと目標ベクトルの内積から、ROV の移動ベクトル (ROV の進むべきベクトル) を算出する。最後に移動ベクトルの方向に進むためのモーター駆動パラメータを ROV に送信し、ROV の自動的な航行ルート制御を行う。方向ベクトルの具体的な求め方は、後述する。

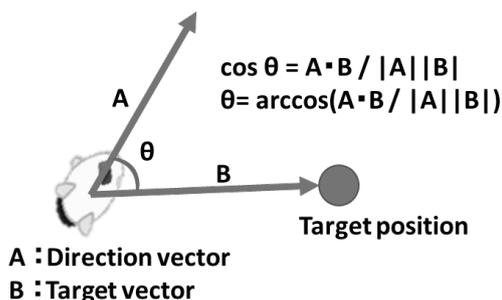


図7 ROV 方向ベクトルの算出方法

4.3. ドローンの位置検出実験

本章では、前述したドローンの位置検出の実験結果について述べる。実験はまず陸上の安定した環境での実験を行った。その後、実際の海洋で実験を行った。

4.3.1. 陸上における X, Y 座標の検出実験

画像からドローンの X, Y 座標を検出する実験を行う。図8に示すようなドローンが黒いライン上をトレースして走行するプログラムを開発した。ドローンは実際に使用する ROV の 30% に縮小したスケールモデルとしている。プログラムの開発はロボット制御のスタンダードである ROS (Robot Operation System) を用いた^[8]。

ドローンの前方下部に装備した4つのフォトレンジスタの値から、黒いラインを検知し、ライン上を走行するようにドローンの速度と方向を制御した。ライン形状は、海中探査におけるサンゴなど海中の障害物を避けることを想定して曲線の多い複雑な経路

とした。走行しているドローンを上空から撮影し、CNNを用いてドローンの検出を行い、検出した矩形の中心座標をX, Y座標とした。以降の実験におけるCNNは、YOLOv3を用いた。

撮影した動画から300フレームごとに画像を取り出した。その画像からCNNを用いてドローンを矩形で検出し、矩形の中心座標(X_i, Y_i)を表1に示す。表1には、実際にドローンがいる座標(X_j, Y_j)も示した。

誤差率は次式で計算した。つまり、誤差率は、図9に示すように、(X_i, Y_i)と(X_j, Y_j)の距離を30%に縮小した調査範囲の一片の距離で割り算した。

$$\text{Error Rate} = \frac{\sqrt{(x_i - x_j)^2 - (y_i - y_j)^2}}{10[m] * 30\%} \quad (7)$$

表1より誤差率は0.3%以下に収まり精度の良い検出が行うことができた。これよりCNNによる物体検出を用いることで、ドローンのX, Y座標の検出が行えることが分かった。

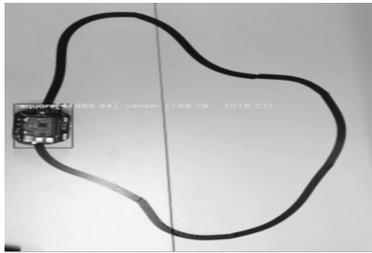


図8 ドローンの走行ルート

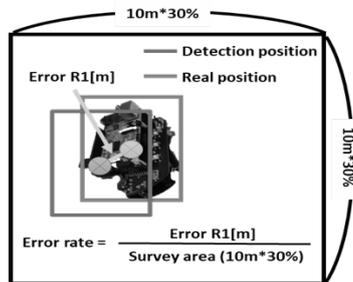


図9 誤差率の算出方法

表1 X, Y座標の検出実験の結果

フレーム番号	実位置(X _i , Y _i)	検出位置(X _i , Y _i)	誤差率
0	(10.16, 51.44)	(9.92, 52.38)	0.3%
300	(7.37, 72.5)	(7.48, 72.44)	0.04%
600	(10.05, 97.75)	(10.20, 97.75)	0.05%
900	(16.34, 105.19)	(16.27, 105.31)	0.06%
1200	(21.52, 105.44)	(21.9, 105.375)	0.1%

4.3.2. Z座標の検出実験

画像からドローンのZ座標を検出する実験を行う。これは、ROVの深度が深くなるにつれて、ROVを検出した矩形の面積が小さくなることを利用する。

ドローンが検出された矩形の面積Sと検出したドローンの高さZ_i、及び実際にドローンがいる高さZ_jを表2に示す。Z_iは次の近似式から求めた。この近似式はSとZ_jを2次多項式で近似した。

$$Z_j = 24,024.43976 S^{-0.5} \quad (8)$$

また、誤差率は次式で計算した。

$$\text{Error Rate} = \frac{|z_i - z_j|}{z_i} \quad (9)$$

誤差率は240mを超えると大きくなった。誤差率が大きくなった原因は、画像に写るドローンの大きさが小さくなり、物体検出に失敗したためである。小さく写ったドローンも教師画像に加えてCNNの再学習を行うことで、誤差率を小さくできる可能性がある。

表2 Z座標の検出実験の結果

検出面積 S[px ²]	実高さ Z _i [cm]	検出高さ Z _j [cm]	誤差率
82592.37	80	83.60	4.50%
36493.43	120	125.76	4.80%
22799.91	160	159.11	0.56%
14739.30	200	197.89	1.06%
8901.53	240	254.64	6.10%
6487.42	280	298.28	6.53%

4.3.3. 水中におけるX, Y座標の検出実験

沖縄水納島にて海中で撮影した画像でROVの検出

を行った。撮影方法は、海上でボートに乗った撮影者が海中にカメラで海底方向へ撮影を行い、ボート下を陸上から操縦する海中ドローンを通させた様子を撮影した。撮影した画像の例を図10に示す

実験結果を表3に示す。実際の位置とドローンAIから出力された位置を1秒ごとに比較した。陸上実験のX,Y座標検出と同じく画像内の実際の海中ドローンの位置を手作業で囲んだ矩形の中心の座標を実位置として、実位置と検出位置の差を想定している調査範囲の調査範囲の一片の距離の10mで除算したものを誤差率とした。

誤差率は0.14%以下に収まり精度の良い検出が行えた。これより陸上で完成させたドローンAIが海洋環境でも十分に位置検出を行え、本システムの有用性が明らかとなった。

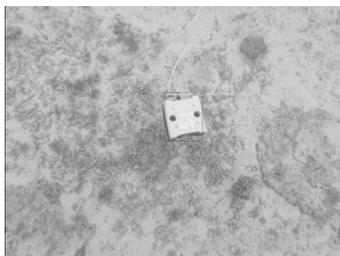


図10 海面から画像検出したROV

表3 X, Y座標の検出実験の結果

フレーム番号	実位置(X, Y)	検出位置(X, Y)	誤差率
0	(109.24, 136.31)	(108.96, 135.54)	0.08%
30	(105.41, 101.69)	(105.99, 101.76)	0.06%
60	(113.73, 64.67)	(114.22, 63.35)	0.14%
90	(146.34, 26.42)	(146.16, 26.90)	0.05%

4.3.4. 位置検出において水の性質が画像に与える影響

上述の報告の中で位置検出が非常に精度の良い結果が得られた。これは、晴天で水質の良好な好条件の環境で行ったため水質によるノイズが少なかったためである。そこで、水質によるノイズが認識率に与える影響を調査した。

Water-Netを用いて、入力した水中画像にホワイトバランスとヒストグラム等化、ガンマ補正の処理

をそれぞれ行った3つの画像を組み合わせ、水質のノイズを低減するAIの導入を行った^[9]。

晴天で水質の良好な好条件の環境で撮影した水中画像と曇天で水質に濁りがある劣悪な環境で撮影した画像を用いて、各フレームのオリジナルの画像とWater-Netから得られる画像をCNNに入力して、検出率を求めた。各画像の例を図11に示す。

各テスト画像における検出率の結果を表4に示す。結果、オリジナル画像とWater-Netで水質を軽減した画像での認識率の変化は変化が少なかった。このことより、水質が画像検出に与える影響は少ないことが分かった。水の濁りが強い場合でも、同様な水の濁りが強い画像をCNNに学習させることで正確な認識が行えることが明らかとなった。

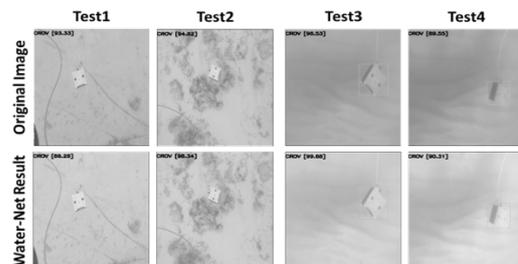


図11 オリジナル画像とWater-Netの結果

表4 オリジナル画像とWater-Netの検出率

	認識率(オリジナル)	認識率(Water-Net)
Test1	88.33	84.78
Test2	92.83	91.94
Test3	90.08	93.54
Test4	93.67	93.75

4.3.5. ドローンの走行ルート制御実験

前節までの実験で画像からドローンの位置(X, Y, Z座標)の検出が可能となった。本章では、ドローンの位置情報からドローンの走行ルートを自動で制御させる実験を行う。なお、本実験では、Z軸方向の走行ルートは制御しないため、ドローンの位置情報としてZ座標は用いない。また、最終的には、ROVを制御する目標とするが、本研究調査助成金を用いた研究開発では、陸上ドローンを制御することで基

本性能を確認することまでに至った。

図12に実験構成を示す。実際の海洋環境では、設備コストと公衆回線のエリアによって、様々なネットワーク形態になることが予想される。そこで、実験構成はドローン側を5G、制御PC側にLTEを使用した非対称な構成とした。

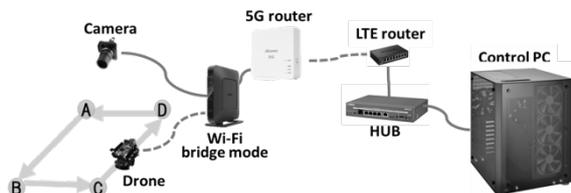


図12 ドローン走行の実験構成

4.3.6. 方向ベクトルの検出方法

本実験でドローンの走行ルートを制御するには、ドローンの位置の他に、ドローンの方向ベクトルも必要となる。図13に示すように、ドローンの前方に張り付けた赤いテープと後方に張り付けた黄色のテープをCNNを用いて矩形で検出し、黄色のテープを検出した矩形の中心座標から赤色のテープを検出した矩形の中心座標に向かう向きをドローンの方向ベクトルとした。

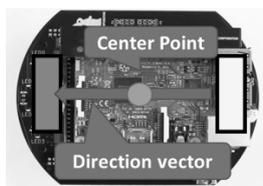


図13 方向ベクトルの定義

4.3.7. 実験結果

図14にドローンの走行ルートとして設定した9個の目標地点を示す。実験では、目標地点Aから出発して

C→F→D→G→I→F→D→A→B→H→I→C→B→H→G→Aの経路でドローンを走行させた。

走行時の様子を図15に示す。①、②、③は10秒ごとのドローンである。ドローンは目標Aに向かった後、向きを変えて次の目標Bに向かって走行している。

図16のように2点の目標を結んだ線分とドローンとの垂線の距離を、調査範囲の一片の距離で除算し

たものを誤差率と定義する。目標のポイント数を9個設定し、これを5周回させ、10秒ごとに誤差率を求めた。

表5に結果を示す。誤差率の平均は1.5%以下となった。この誤差率が海中生物調査にとって十分な精度であるかは、現時点で不明であるが、今後、専門家へのヒヤリングなどで明らかにする。本実験ではドローンはルートを大きく外れることなく制御可能であったことから、提案手法の効果が示されたと考えている。

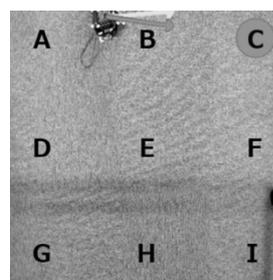


図14 走行ルート

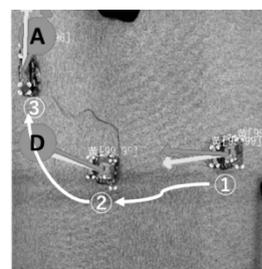


図15 走行ルート自走制御の様子

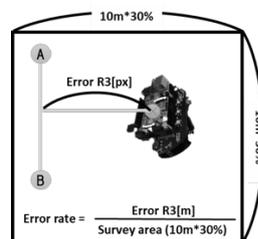


図16 誤差率の算出方法

表5 ドローンの走行ルート制御実験の結果

	1周目	2周目	3周目	4周目	5周目	平均
ルート誤差率	1.90%	1.72%	1.39%	0.86%	1.75%	1.52%

5. ネットワーク特性を含めた ROV 自動航行実験

5.1. 実験内容

海中生物調査システムは、水上ブイと陸上の Control PC の間に通信ネットワークを用いる。水上の通信は波による電波の反射や基地局からの距離が遠いこと、海上は障害物がなく見通しがよいため他の電波との干渉が生じやすいためネットワーク障害が起きることが報告されている^[10]。

本実験では、ネットワーク障害としてパケット遅延に着目する。パケット遅延によって、水中ドローンの自動航行の精度が変化する。パケット遅延が大きくなると、水中ドローンに与える制御情報の時間間隔が長くなり、水中ドローンがルートから外れ、ルートに対して蛇行する可能性がある。

図 17 の実験構成に、水上でのネットワーク障害の再現として、水中ドローン側のネットワークにパケット遅延を加えることが可能なネットワークエミュレータを挿入して実験を行う。ネットワークエミュレータで 3D シミュレータから Control PC(上り)方向と Control PC から 3D シミュレータ(下り)方向でパケット遅延を挿入し、水中ドローンの自立航行実験を行う。

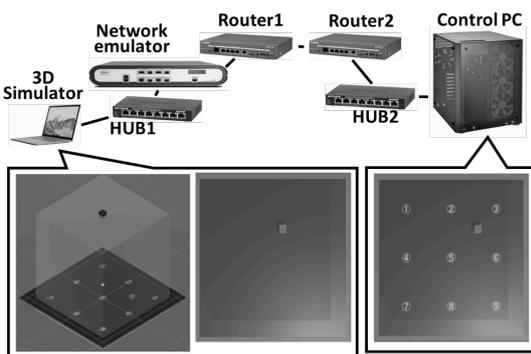


図 17 ネットワークの遅延を含めた実験構成

5.2. 実験結果

図 18 は水中ドローンが走行した軌跡である。図 19 は水中ドローンの深さ距離を変化させたときのルート誤差率 P_{root} の平均値である。ルート誤差率 P_{root} は上り方向と下り方向ともに 2.6%以下であった。3000[ms]でルート誤差率 P_{root} が下がっているのは、制御情報の遅延により水中ドローンの角度 θ_{drone} が大きくなり $\theta_{th} < \theta_{drone} \leq 180^\circ$ を満たしたことで、水

中ドローンが頻繁に曲がったためである。

上記の結果より、パケット遅延が小さい時は制御しきい角度 θ_{th} を小さく設定し、パケット遅延が大きい時は制御しきい角度 θ_{th} を大きく設定する。つまり、ネットワークで発生するパケット遅延に応じた制御しきい角度 θ_{th} を選択することで、水中ドローンのルート誤差率 P_{root} を低減できることが分かった。

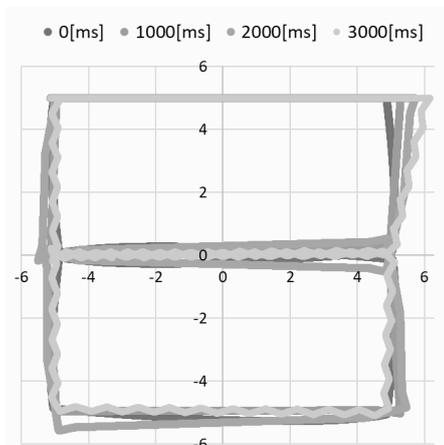


図 18 遅延を変化させたときの走行の軌跡

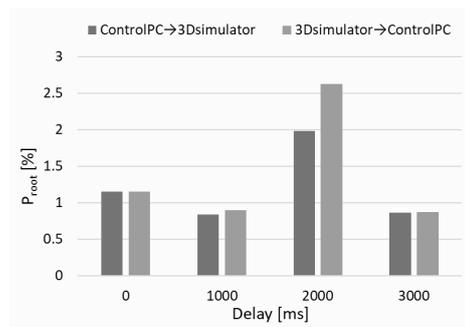


図 19 自動制御のルート誤差率 P_{root}

発表論文

- [1] Yuta Kogi, David Pich, Tatsuhiko Nishi, Hiroki Kamehama, Kastuya Nakahira, Novel Unmanned Marine Life Survey System using Image Detection Techniques based on Neural Network, MDPI, sustainability, 2023年10月(予定)

口頭発表, 受賞等

- [1] Yuta Kogi, Hiroki Kamehama, Katsuya Nakahira, Study for an Unmanned Marine Life Survey System using Image Detection Techniques, 第1回高専研究国際シンポジウム, 2022
- [2] 近木 裕太, 中平 勝也, 海洋プイと海中ドローンを組み合わせた海洋生物調査システム, 6th Okinawa Marine Science Workshop, 2022
- [3] 近木 裕太, 中平 勝也, 海洋プイと海中ドローンを組み合わせた海洋生物調査システム, 知覚情報研究会/次世代産業システム研究会, 2022
- [4] 西 達大, 亀濱 博紀, 中平 勝也, 美ら海における水中ドローンの自立航行手法の提案, 動的画像処理実利用化ワークショップ, 2022
- [5] 西 達大, 中平 勝也, 画像認識と音響測位を組み合わせたROVの航行ルート自動制御手法, 動電気学会 電子・情報・システム部門大会, 2021
- [6] ピッチ デイビット, 亀濱 博紀, 中平 勝也, Improving the Performance of Deep Learning-based Method for Underwater Image Enhancement by Fusing Known Features from Spatial-Domain Method, 動的画像処理実利用化ワークショップ, 2021

参考文献

- [1] 環境省: "モニタリングサイト1000", <http://www.biodic.go.jp/moni1000/index.html>
- [2] 大垣俊一: コドラートとトランセクト: 海岸調査の精度論, 関西海洋生物談話会の連絡誌, Argonauta 16: 16-24, 2009, <http://www.mus-nh.city.osaka.jp/iso/argo/nlindex.html>
- [3] 中平 勝也, ピッチデイヴィット: 画像認識を用いたマリンレジャーサポートシステムの研究開発, OS5-3, 2019
- [4] Joseph Redmon, Ali Farhadi: YOLOv3: An Incremental Improvement, CoRR, 2018
- [5] 除潤東, 藤野巖: 機械学習を用いた画像識別手法の魚の種類認識への応用, 情報処理学会第82回全国大会, 2020
- [6] 国土交通省: 平成27年度 次世代社会インフラ用ロボット

(水中維持管理用詳細版), 2015

<https://www.mlit.go.jp/common/001110595.pdf>

- [7] 稲葉祥梧, 岡本章裕, 瀬田剛広, 今里元信, 篠野雅彦, 金岡秀, 田村兼吉, 渡邊佳孝, 松本宙: AUV 駆動音による水中音響通信・測位装置への影響評価について, 第16回海技研究発表会講演集, 2016
- [8] 上田 隆一: Raspberry Pi で学ぶ ROS ロボット入門, 日経BP, 2017
- [9] Chongyi Li, Chunle Guo, Wenqi Ren, Runmin Cong, Junhui Hou, Sam Kwong. Dacheng Tao: An Underwater Image Enhancement Benchmark Dataset and Beyond, arXiv
- [10] 国土交通省: "海上における通信環境について(参考資料)", 第113回 船員部会 配布資料, 2019, <https://www.mlit.go.jp/common/001299753.pdf>