



沈モビリティ

ツツ子

機体紹介

メンバー：

福田 雄
十亀 孝彰

小林 優希
森本 拓馬

機体スペックと構成

全長[mm]	900
全高[mm]	300
全幅[mm]	400
重量[Kg]	6.1
スラスタ数	4
プロセッサ	RaspberryPi 4B
バッテリー	LiPo 14.8V 6500mAh

カメラと超音波センサから風船の位置を測定し、アームを使って保持することで、確実に風船を割ることを狙う。

機体フレームにはアルミフレームを採用し、スラスタやセンサ等、モジュールの配置変更を容易に行う事ができる。

横揺れ方向の安定には機体の復原力を生かし、スラスタ数を削減した

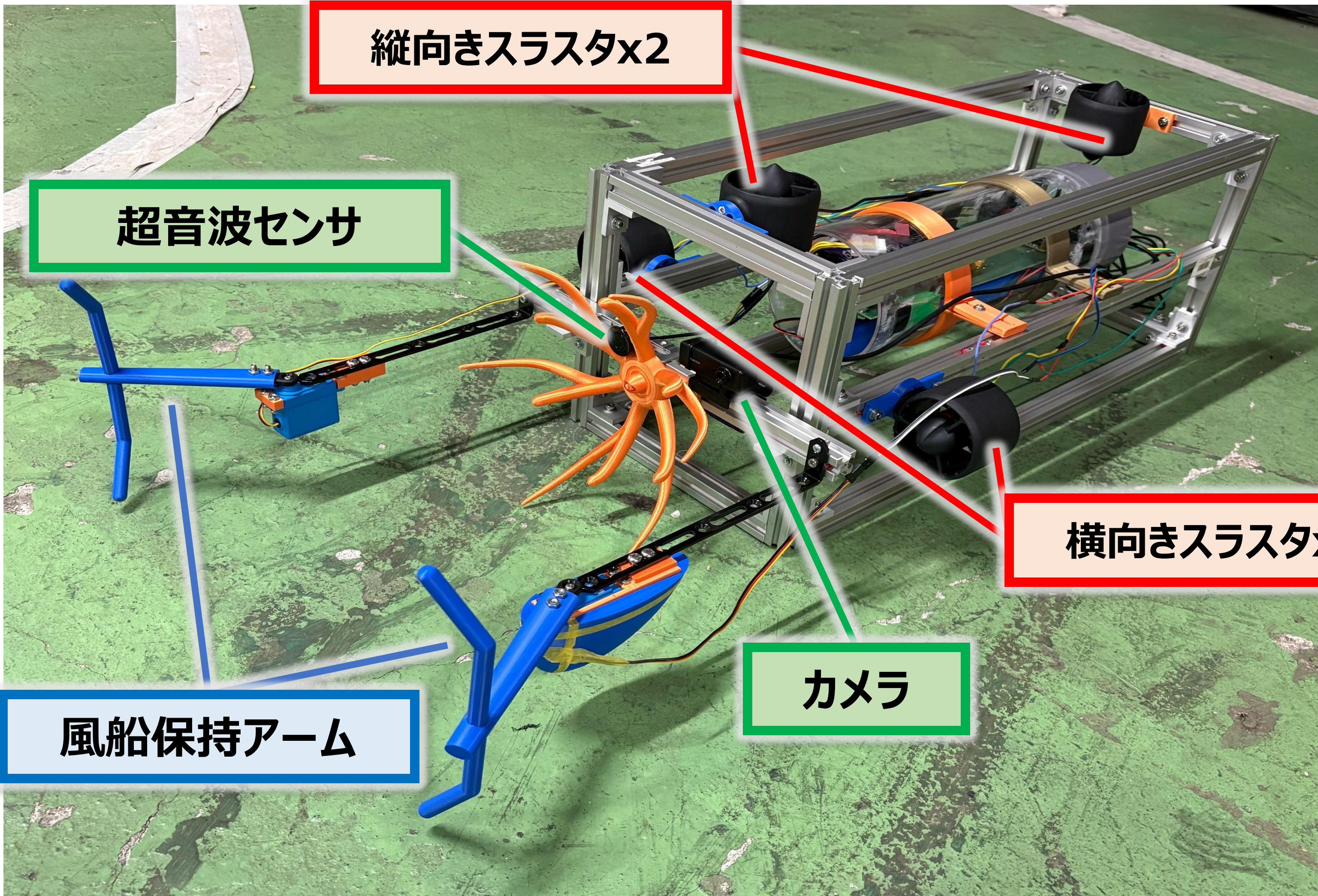


図1 開発中の機体

※資料提出段階の構成

シミュレータの活用

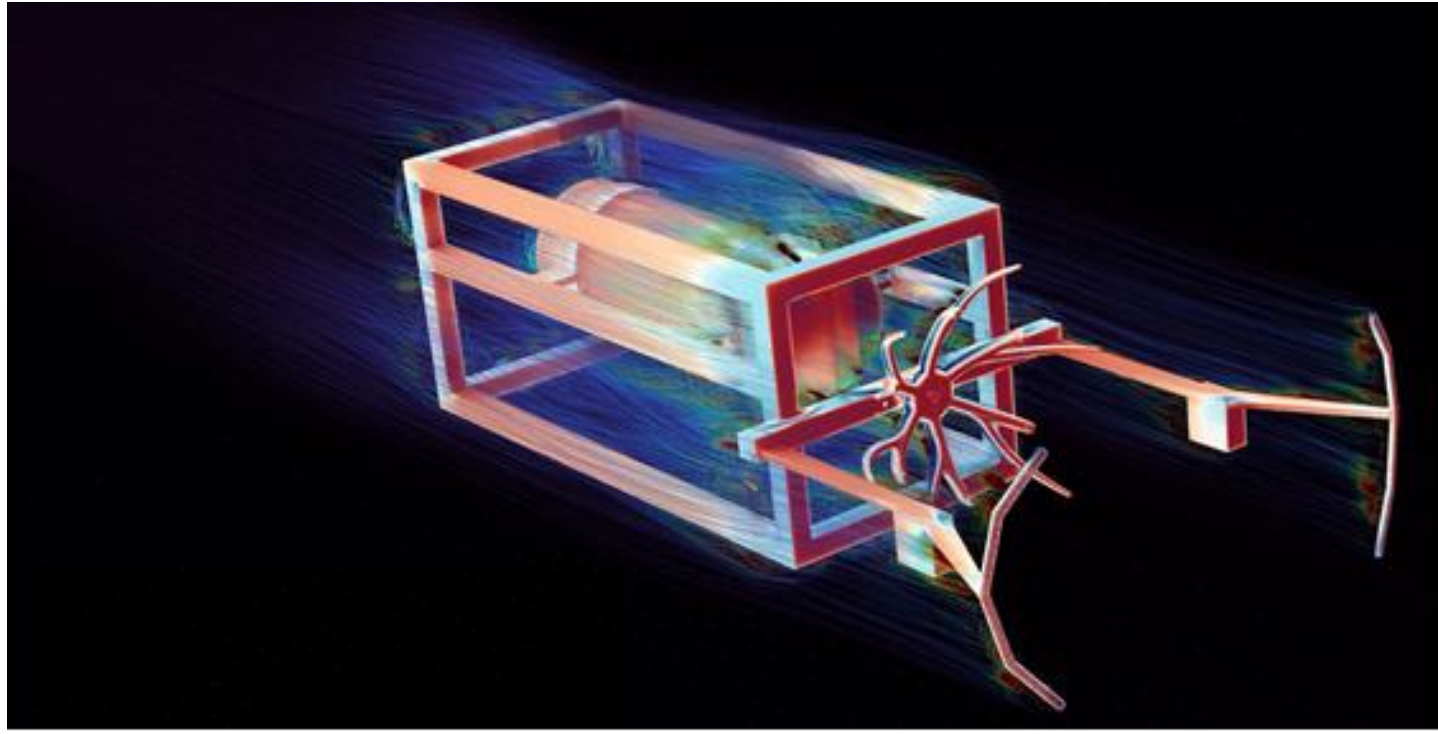


図2 CFD解析例

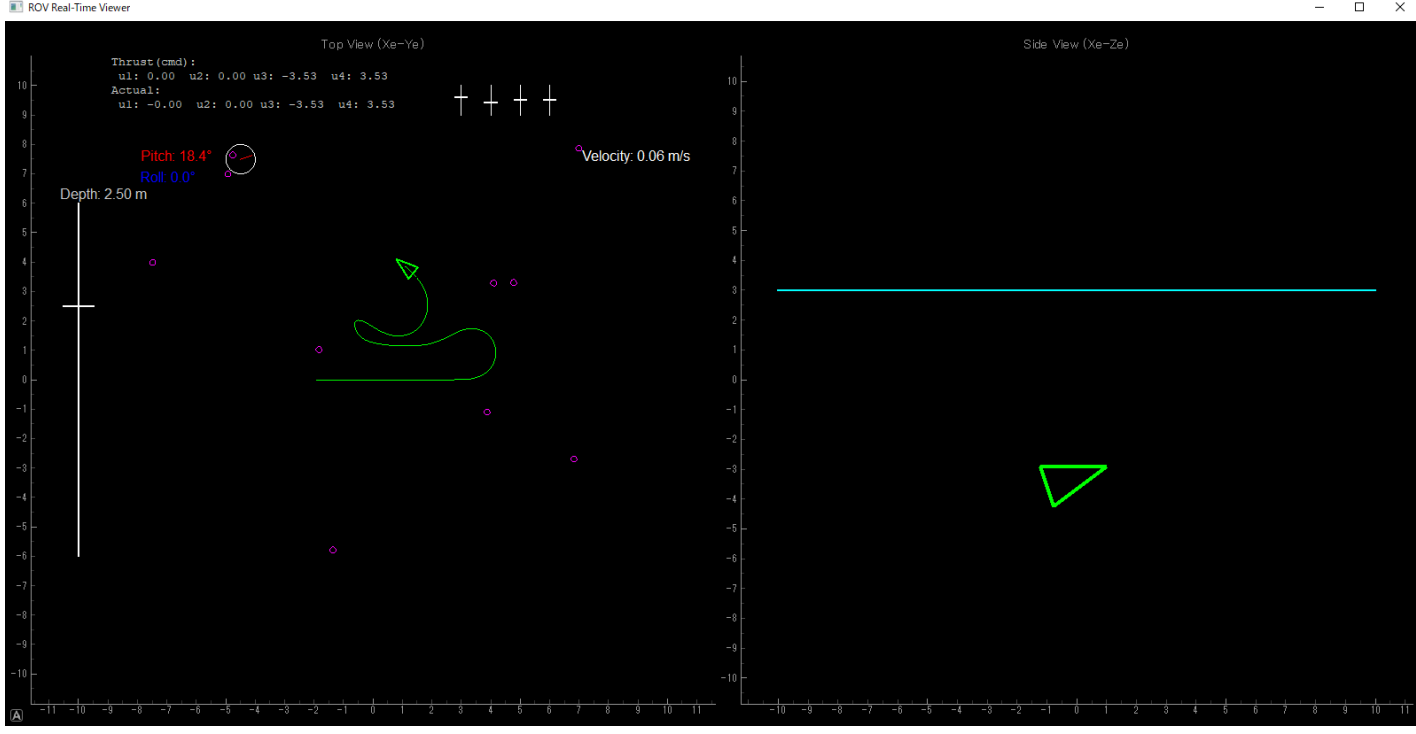


図3 6DoF運動モデルによる自作シミュレータ

CFD解析や自作シミュレータを用いておおその制御器の設計を行った。最終的な微調整は実機試験にて~~手~~合いで行われる。(KID制御)

風船保持部の製作

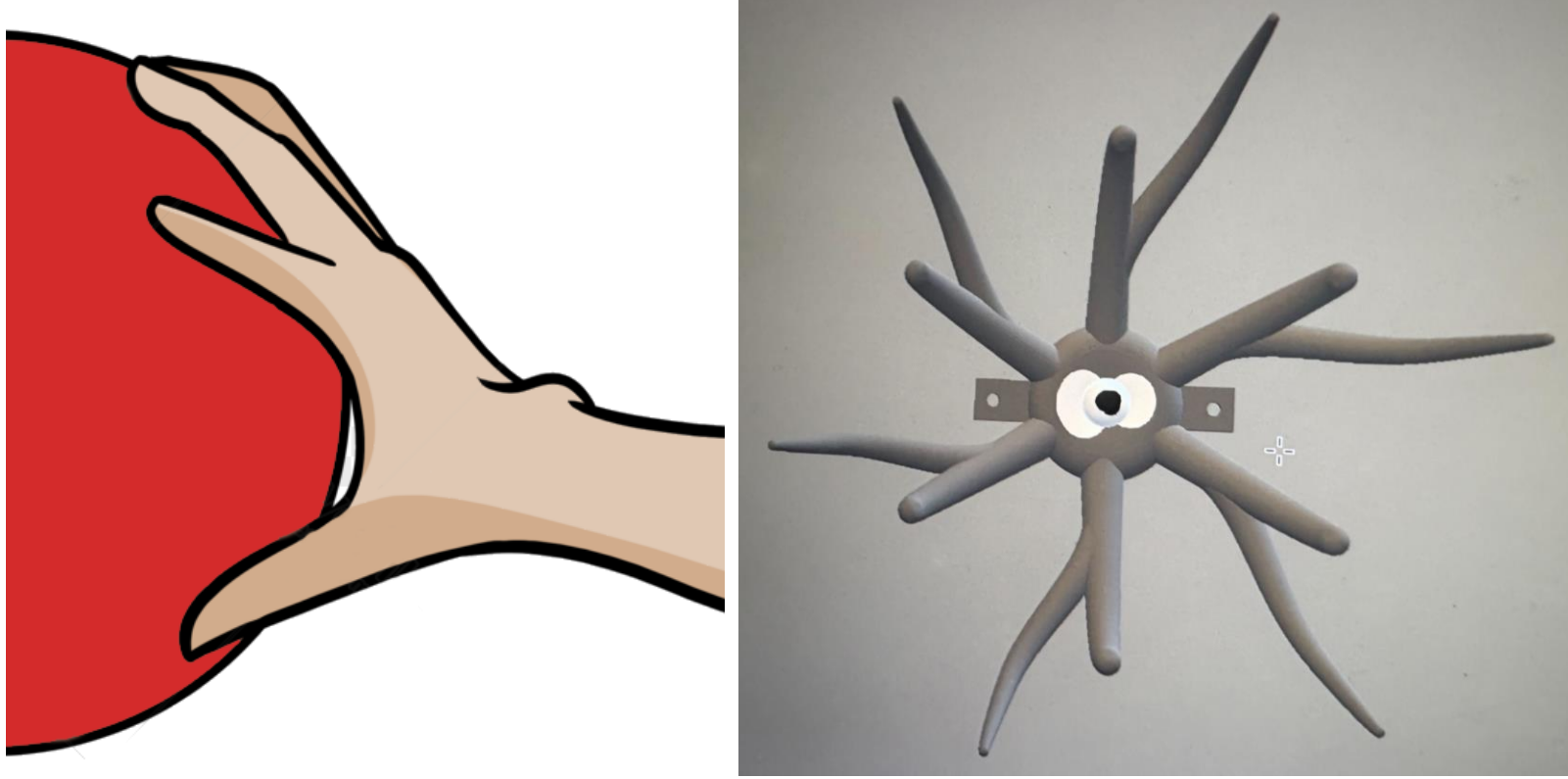


図4 風船を把持する手をイメージした有機的設計



図5 3Dプリンタによる造形

アピールポイント！

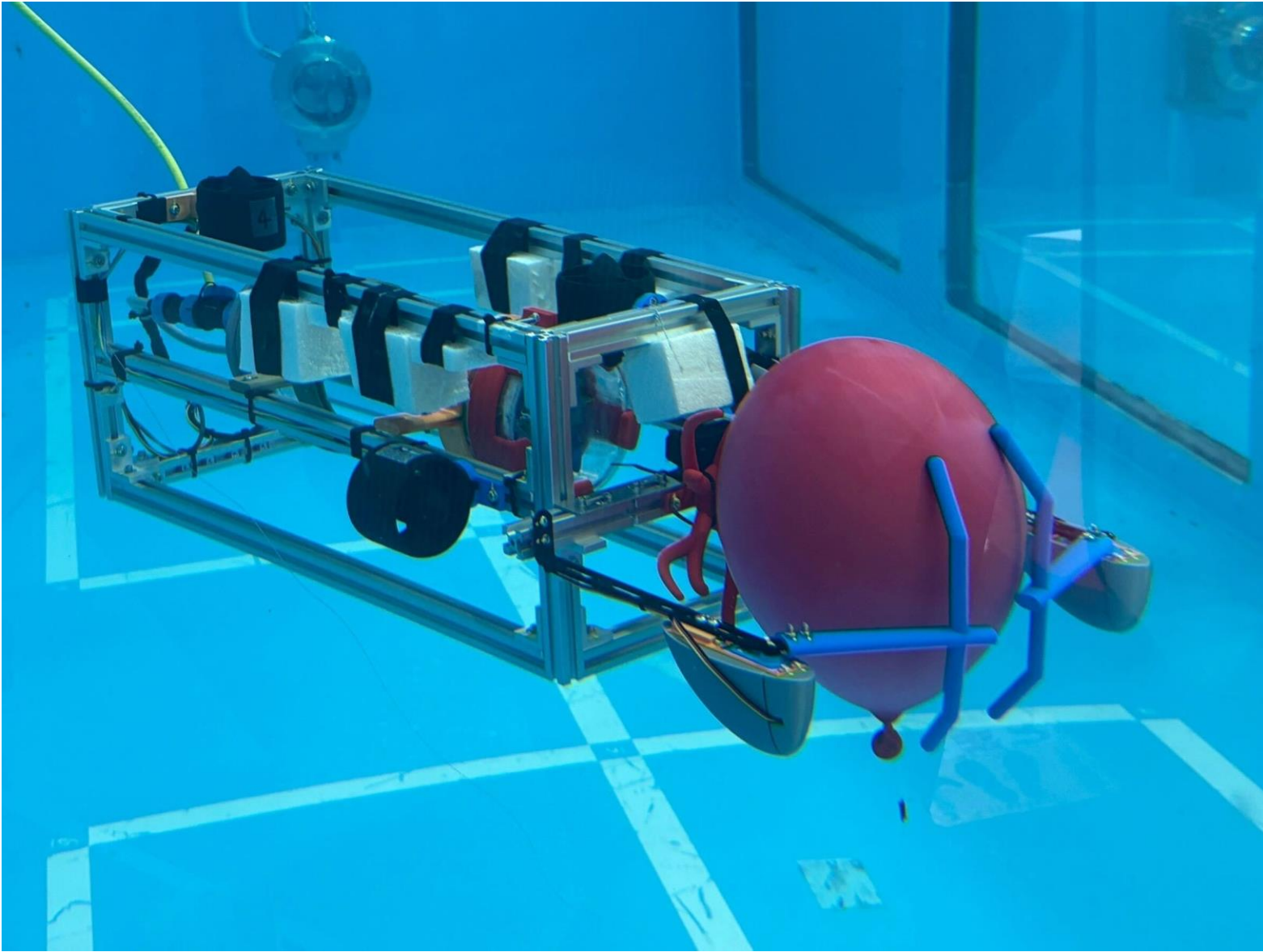


図6 水中での風船捕獲試験

過去年度の映像から風船が動いてしまうことが割ることの障害になっていると考察。風船を確実に割るため、アームでの風船保持機構を搭載。

- 1. 水中で風船を捕獲
- 2. 流れによる動揺に対しロバスト

という2つのコンセプトのもと設計、製作を行った

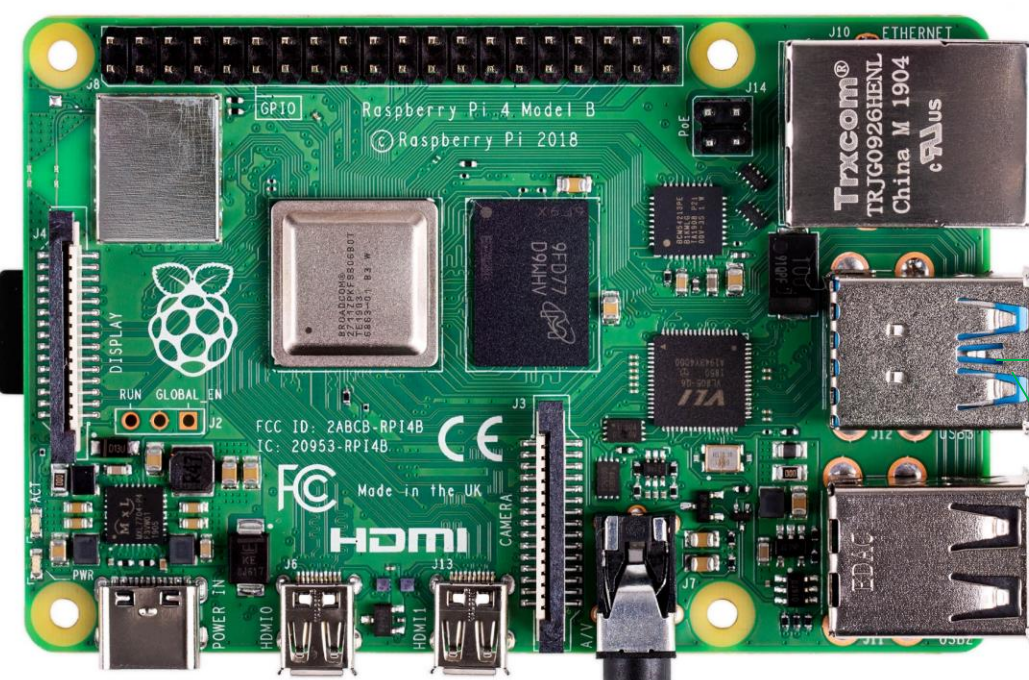
ソフトウェア

○Python分散処理フレームワーク【Ray】を用いたコア分割



RAY

- ・機械学習などに用いられるフレームワーク
- ・コアの分割がしやすく、分散処理が容易に実現可能



センサデータ処理

- ・深度センサ
- ・IMU
- ・超音波センサ

カメラ画像処理

- ・OpenCV
- ・2値化処理

モジュール制御

- ・スラスタ
- ・サーボ

コアを分割し、並列で処理を行うことでリアルタイム性を確保

○OpenCVを用いた画像処理



図7 カメラでの風船認識の様子
(水中ロボットコンベンション公式Youtubeの動画を使用)

【赤色の風船を識別するまでの4ステップ】

1. カメラで取得したRGBデータを領域で平均化を行いノイズ処理
2. RGBデータをHSVデータに変換
3. 色相から閾値処理を行い赤色を特定
4. 領域の面積閾値を設けて、風船を識別