・目的

グライダーの速度計は全圧と静圧の差から求めるため、実際の速度と速度計の速度でタイムラグが生じる。よってグライダーを操縦する際は、速度計を頼りにするのではなく地平線の見え方を頼りにしている。そのことからセンサーではなく、画像処理によって制御する方法を思いついた。

今回のプロジェクトでは OpenCV によってシミュレータの映像を画像処理し、地平線を認識することでグライダーの姿勢を読み取り、自動で適切なピッチになるように操作することを目標とした。

・計画

本プロジェクトは大きく 3 段階に分けることができる。 1 段階目はシミュレータの映像を画像処理するプログラミングを作ることである。2 段階目は画像処理の結果からグライダーの姿勢を読み取ることである。 3 段階目はグライダーを適切な姿勢にするためにシミュレータを操作することである。

・調査方法

画像処理に関しては OpenCV を用いた。具体的な方法としてはシミュレータ映像内の長い線を地平線として検出した。姿勢は、実際の操縦と同じようにキャノピーに取り付けられたひもと地平線までの距離をもとに推定した。ひもも OpenCV を用いた画像処理によって検出した。シミュレータの操作はピッチを操作するボタン（パソコンの↑ボタン、↓ボタン）を押せるロボットを制作した。

・活動経過

7 月　シミュレータのスクリーンショットの地平線を検出する。

8 月　シミュレータ映像の録画動画を画像検出する。

8 月　シミュレータ映像を画像検出する。

9 月　ロボットを制作する

・成果

地平線の検出は gray スケールにして、2 値化した画像を Hough 変換することで線を検出した。ひもの検出はヒストグラム平坦化した画像を HSV 空間に移行して赤色の閾値でマスクした。検出したひもの位置と地平線までの距離を点と直線の距離の公式を用いて計算した。検出する画像の読み込みはリアルタイムでの録画とデータの転送ができる OBS studio というアプリを使用した。ボタンを押すロボットはレゴスパイクを用いて制作した。プログラミングに関しては SPIKE Prime という専用のアプリがあったが OpenCV の画像検出と同時に実行しなければならないため、Visual Studio Code を用いて行おうと思った。そこで SPIKE-RT という SPIKE Prime 向けソフトウェアプラットフォームを用いた。しかし SPIKE-RT は C 言語を用いているが OpenCV は C 言語を用いてプログラミングすることができなかった。