JAIST Repository
https://dspace.jaist.ac.jp/

Title	拡張現実に基づく視覚的なドローン操縦の実現
Author(s)	平山，研太郎
Citation	
Issue Date	2020-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/16341
Rights	
Description	Supervisor:宮田　一乗，先端科学技術研究科，修士（知識科学）

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

修士論文

拡張現実に基づく視覚的なドローン操縦の実現

平山　研太郎

主指導教員　　宮田　一乗

北陸先端科学技術大学院大学
先端科学技術研究科
（知識科学）

令和２年３月

Abstract

The drone is utilized in various applications, such as the inspection of a building, pesticide dispersion for agricultural purposes. A drone can fly in any direction under control. However, it is complicated to control in the far away position with poor recognition. Therefore, various approaches are proposed to save the labor of automation control as complicated operations.

Global Positioning System (GPS) can be used to measure the positions of a drone so that the drone can fly in the appointed routes. However, the electric wave of the GPS signal wave is interrupted among the buildings. A drone cannot receive the GPS electric wave from the satellite in indoor environments.

The previous work can fly the drone by recognizing a marker placed on the floor and landing on it with a mounted camera. The commercial products can not only land the drone on a marker and, make a turn around the marker, and so on. Although the drone can fly automatically even in places where GPS radio waves cannot be received, it is difficult to place the markers in some specific situations. Moreover, the drone may fail to fly following the markers due to external factors such as wind.

The previous method could simulate the flight and reconstruct the flight location in virtual three-dimensional space. The flight is designed by drawing the path in virtual space or decided by recognition of the obstacle by the camera. Although this a technique allows visual control, it cannot be used in places where 3D reconstruction has not been well built.

To address these issues, this thesis proposes a system that flies the drone following the generated object in the virtual space. In addition, the path of the flight is allowed to check with visualization result. Furthermore, we aim to simplify the design interface for the flight path authoring touching on a tablet.

Procedure of the proposed system;

(1)The drone recognition using the iPad camera in reality.

(2)The placement of the object A at the location of the recognized drone in the virtual space.

(3)The placement of the object B at the location of the destination where the user expected by pointing with the iPad camera.

(4)The generation of the object C at the real-time position of the drone which chasing from start point(object A) to the destination(object B) automatically in virtual space.

(5)Automatically flying the drone by synchronizing the position of the drone following object C in real-time.

We tested that the proposed system can generate object A at the same position as the drone once the iPad camera pointing at the drone.

We describe an example of the object B that the user places in the virtual space as a path as follows.

Firstly, a spherical object that moves in a circle is generated. We considered that the circular flight is intending for pesticide spraying. The drones are expected to be used as a supplement to solve the serious labor shortage in the security industry. Similarly, in order to meet the labor shortages in the agricultural purpose, measures are being considered to expand the use of the drones.

Secondly, we create a rectangular parallelepiped object as a wall. We considered that flying along the wall is a movement for building inspection. The drones are expected to be used for inspection of bridges and tunnels to reduce the number of tasks.

Finally, we created objects with irregular shapes and movements. We assumed that the user is composing the movement of the camera and the movement of the 3D object when creating a CG movie.

There is a conventional method called match move. In this method, the user infers the moving quantity of the camera from the picture of the live-action film and compose the 3D object. Because the user is responsible to match it with the movement of the camera, the CG designers are not able to design the movement freely.

The proposed system aims at the flight of drone can deal with any kinds of three-dimensional CG object movements. As a result, there is no limitation for designing the movement of the CG object by the designers.

When object C changes its direction of movement in the virtual space, the proposed system will automatically send a command to the drone to modify its flight according to the changing.

We suggested using the object C movement in virtual space to control the drone movement in reality. The proposed system allows the user to fly a drone in the situation that GPS could not use, the space where has not be rebuilt into the 3D environment on the computer or where the markers are difficult to place.

The proposed method also allows users to fly the drone by touch operation while viewing the augmented reality space from the tablet computer. As a result, even a person who has difficulty in manual control of the drone can easily specify an arbitrary flight route.

目次

第 1 章 はじめに 1

1.1 研究背景 1

1.2 研究目的 2

1.3 論文構成 2

第 2 章 関連研究 3

2.1 三人称視点からの操縦に関する研究 3

2.1.1 3
2.1.2 4

2.2 自動的な操縦に関する研究 5

2.2.1 5
2.2.2 6

2.3 本研究の位置づけ 7

第 3 章 提案手法 8

3.1 使用する機材とコンピュータソフトウェア 8

3.1.1 ドローン 8

3.1.2 拡張現実 9

3.1.3 モデリングとアニメーション 10

3.1.4 統合開発環境 10

3.2 システムの構成 11
第 4 章 システムの構築 12
4.1 物体認識 12
4.1.1 物体のスキャン 12
4.1.2 物体認識完了状態の表示 13
4.2 3D オブジェクトの製作 14
4.2.1 円状に移動する球体のオブジェクト 15
4.2.2 壁面を模した直方体 16
4.2.3 不規則な形状と動きを持つオブジェクト 17
4.2.4 オブジェクトの配置 18
4.3 ドローンの自動的な飛行 19
4.3.1 追跡するオブジェクト 19
4.3.2 当たり判定 20
4.3.3 コマンド送信 21
4.4 複数方向への同時操縦 22
4.4.1 この章のまとめ 23
第 5 章 システム構築 24
5.1 GUI 24

5.2 円状に飛行するモード 25

5.3 壁面に沿って飛行するモード 26

5.4 複雑な動きに合わせて飛行するモード 27

第 6 章 実験・評価 28

6.1 実験の環境 28

6.2 実験の目的 28

6.3 実験の内容 29

6.3.1 実験 1 29

6.3.2 実験 2 30

6.4 計測したタイム 31

6.4.1 実験 1 で計測したタイム 31

6.4.2 実験 2 で計測したタイム 32

6.5 アンケート調査 33

第 7 章 実験の結果と考察 36

7.1 実験の結果 36

7.1.1 実験のタイムについて 36

7.1.2 アンケート調査の結果について 36

7.2 考察 37

第 8 章 課題と展望 38

8.1 まとめ 38

8.2 課題 38

8.3 展望 39

第 9 章 謝辞 40

図目次

図 1　ジョイスティックによるドローンの操縦 1

図 2　床に配置されたマーカーの周囲を飛行するドローン 2

図 3　3 人称視点からドローンを確認する。 3

図 4 頭部装着ディスプレイで 3D 空間を視認しながら操縦する 4

図 5 撮影したい物体を指で丸く囲む操作 5

図 6 精細な 3D モデルをドローンの撮影によって作成する 6

図 7　Ryze Tech 社製 TELLO IRON MAN EDITION 8

図 8 ARKit により 3D オブジェクトを現実空間へ配置する様子 9

図 9 Autodesk 社製 MAYA 10

図 10 Unity Technologies 社製 Unity 10

図 11 システムの概要 11

図 12 ARKit Scanner で特徴点を検出する 12

図 13 ドローンを物体認識した位置にオブジェクト A を表示する。..... 13

図 14 円状に移動する球体のオブジェクト 15

図 15 オブジェクトの調節 15

図 16 壁面のオブジェクト 16

図 17 上下運動しながら横方向へ移動する球体..... 16

図 18 オブジェクトの製作過程..... 17

図 19 オブジェクトの動き 17

図 20 オブジェクト B は平面検知した場所の内一か所をタッチして配置する 18

図 21 オブジェクト C に追跡されるオブジェクト B..... 19

図 22 オブジェクト C と随伴するオブジェクト群 20

図 23 ドローンのコマンド(TELLO SDK[11])..... 21

図 24　Cf が 2 枚の板(Cu,Cr)と同時に接触している 22

図 25 オブジェクト B を追跡するオブジェクト C と随伴するオブジェクト群 23

図 26　GUI 24

図 27 自動的に円状に飛行するドローン 25

図 28 壁面に沿って飛行するドローン 26

図 29 不規則な動きに合わせて飛行するドローン 27

図 30 オブジェクトに設定されている関節..... 27

図 31 実験場所..... 28

図 32 バスケットコート 29

図 33　実験 1 の様子 29

図 34 体育館の壁面 30

図 35 実験 2 の様子 30

図 36 本研究に関するアンケート 33

表目次

表 1 実験環境 28

表 2 タイム(単位:秒) 31

表 3 タイム(単位:秒) 32

表 4 アンケート回答(設問①,②) 34

表 5 アンケート回答(設問③) 35

表 6 設問③における回答の平均値 35

第1章 はじめに

1.1 研究背景

現在、ドローンは映像制作、建築物の点検、農薬散布など様々な事例において活用されている[1]。ドローンのあらゆる方向へ自由自在に飛行可能な点は長所である。しかし、操縦者が向いている方向とは異なる上下、前後、左右移動の方向を同時に認識しながら、視認することが難しい離れた位置で操縦することが複雑である点は短所である(図 1)。その為、複雑な操縦を自動化することで省力化する様々な手法が考案されている。

図 1　ジョイスティックによるドローンの操縦

Global Positioning System(以降、GPS)よってドローンの自己位置を計測し、指定した座標の経路を飛行する方式がある[2]。しかし、GPS は衛星からの電波を受信できない屋内や地下、または屋外であっても電波が遮断される高い建造物に囲まれた場所では使用できない[3]。

ドローンに搭載されたカメラで床に配置したマーカーを画像認識する方式では、マーカーの場所へ自動的に着陸させる研究[4]がある。着陸させるだけではなく、マーカーの周囲を旋回させる機能を備えた市販の製品もある(図 2)。GPS の電波を受信できない場所でもドローンを自動的に飛行させることが可能であるが、マーカーを配置することが困難な場所や状況では使用できない。また、ドローンが風などの外的要因によって飛行場所から外れてしまい、マーカーを見

失う恐れがある[5]。

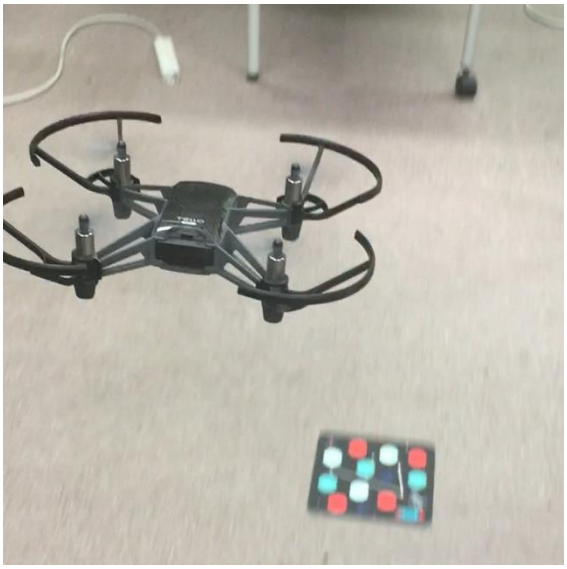

図 2　床に配置されたマーカーの周囲を飛行するドローン

コンピュータ上でドローンの飛行をシミュレーションする方式では、飛行する場所を 3 次元再構成する。その空間に飛行経路となる線を描き、障害物やドローンのカメラで撮影する物体と適切な距離を保ちながら飛行させる[6]。視覚的な操縦が可能な手法であるが、3 次元再構成されていない場所では使用することができない。

1.2 研究目的

本研究では、拡張現実の仮想空間に生成したオブジェクトの動きにドローンが追従し、いかなる場所においても自動的に飛行させることを目的とする。また、オブジェクトの表示によって飛行経路を可視化する。さらに、タブレット型コンピュータを使用したタッチ操作のみでドローンが飛行する場所を指定し、操縦を簡略化することで、ユーザビリティの向上を目指す。

1.3 論文構成

本論文は全 9 章で構成する。第 2 章では関連研究を紹介し、本研究の意義を示す。第 3 章で提案手法について説明し、第 4 章でシステムの構築、第 5 章で試作したシステムの解説をする。第 6 章では実験・評価、第 7 章では実験の結果と考察を述べる。第 8 章で課題と展望を検討する。第 9 章は謝辞である。

第2章 関連研究

本章ではドローンを操縦する際の視点や操縦を自動化することについての関連研究を示す。

2.1 三人称視点からの操縦に関する研究

2.1.1

Ryotaro ら[7]は、ドローンを操縦する際の視点が機体に搭載されているカメラからの一人称視点では、飛行している場所の状況認識を欠くと指摘する。解決策として、二機目のドローンを操縦しているドローンと一定の距離を保ちつつ自動的に飛行させ、その周辺を撮影することで三人称視点からの操縦を実現する手法を提案した(図 3)。三人称視点はドローンの模型を動かす簡単な操作で視点を変更しながら、進行方向の確認や障害物を確認することで回避が容易になるとしている。

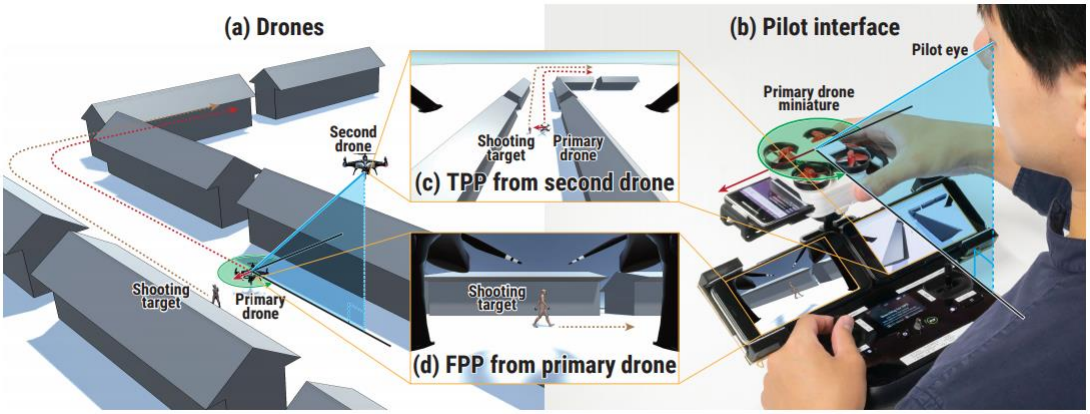

図 3　3 人称視点からドローンを確認する。

2.1.2

Okan ら[8]の研究も三人称視点からドローンを操縦しており、閉鎖的な空間で安全な飛行を実現する手法を提案している。ドローンが X 線を使用して障害物を検知しつつ SLAM（Simultaneous Localization and Mapping）技術で空間を観測した後、3D モデル化することにより仮想現実を構築する。操縦者は頭部装着ディスプレイで構築された空間を視認しつつ、その空間でドローンを模した 3D オブジェクトを移動させることで、現実空間におけるドローンの飛行経路とする(図 4)。

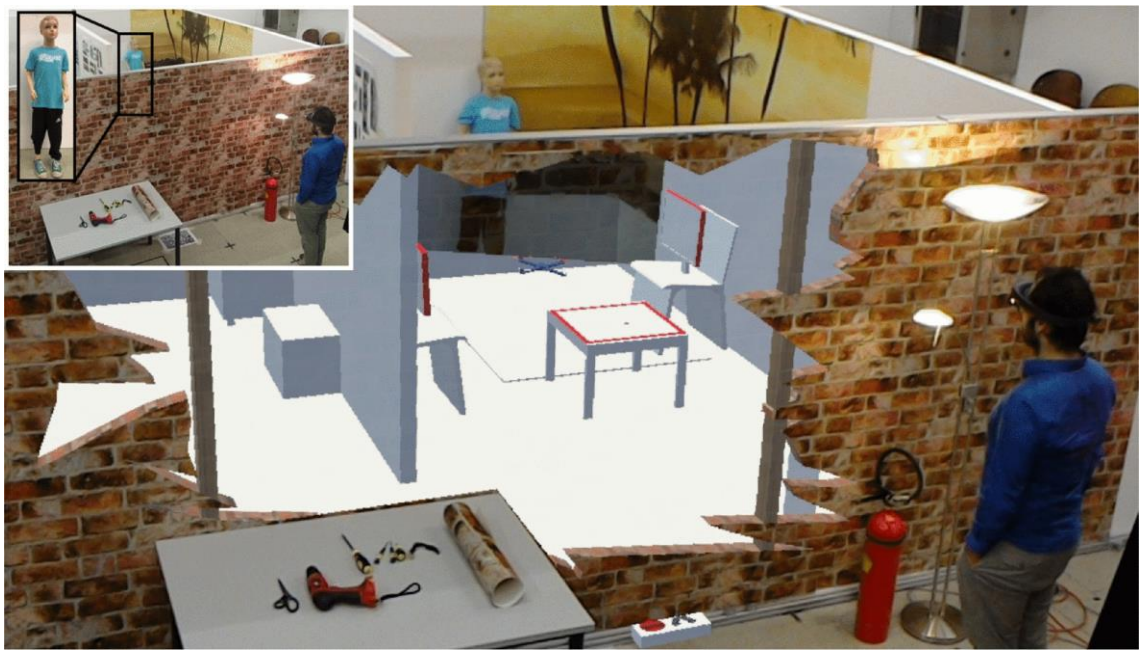

図 4 頭部装着ディスプレイで 3D 空間を視認しながら操縦する

2.2 自動的な操縦に関する研究

2.2.1

Ziquan ら[9]の研究はドローンによる空撮の自動化を提案している。操縦者が撮影したい物体に対して、適切な距離や映りの良いアングルへドローンを手動で移動させることは困難である。その課題に対して、カメラに映った物体の特徴点を SLAM 技術により取得し、物体とその周辺の 3D マップを作成する。3D マップの情報から物体を撮影する飛行経路や視点を割り出し、ドローンを自動的に飛行させる。操作はタブレット型コンピュータで行い、ドローンのカメラに映る撮影したい物体を、指で丸く囲む簡単な方法である(図 5)。

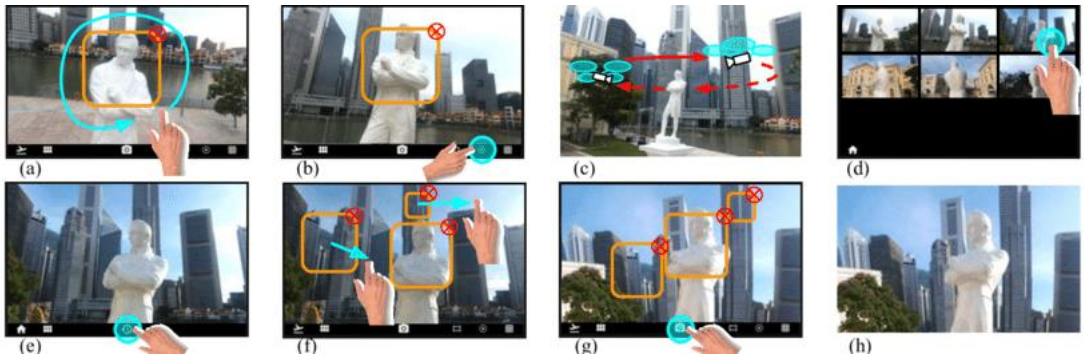

図 5 撮影したい物体を指で丸く囲む操作

2.2.2

Benjamin ら[10]の研究は、ドローンで空撮した画像による建築物の精細な 3D モデルを生成する手法を提案している。まず、操縦者が衛星地図上で撮影する建築物の領域と高さを指定する。その範囲内にある建築物の上空をドローンが自動的に飛行し、その建築物の占有空間とその周囲にある障害物をコンピュータ上に記録する。記録した情報から撮影に適した場所と効率的な飛行経路を割り出した後、2 回目の自動的な飛行をドローンにさせる(図 6)。1 回目の飛行経路で障害物や建築物の位置関係を割り出しているので、2 回目の飛行経路は 1 回目の飛行と比較して、建築物に対してより低空かつ近距離で飛行することができる。このことにより、正確な 3D モデルの生成に必要である詳細な画像を空撮することが可能である。

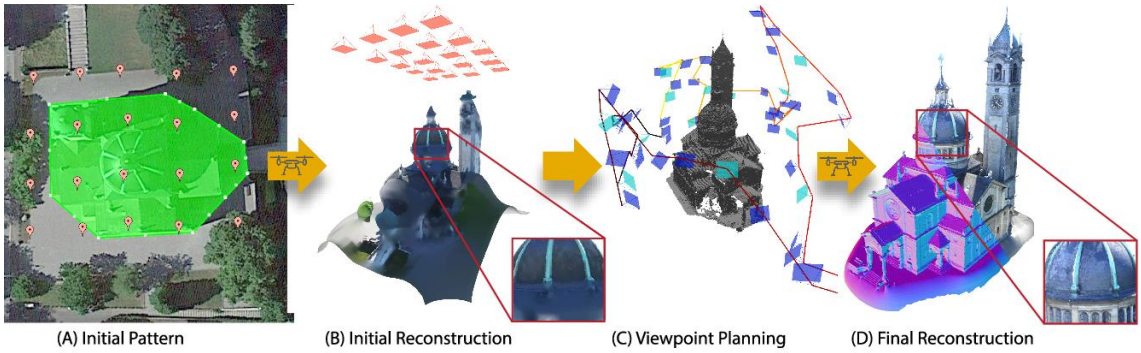

図 6 精細な 3D モデルをドローンの撮影によって作成する

2.3　本研究の位置づけ

本研究では先行研究と比較して、容易かつ迅速に、いかなる場所においてでもドローンの自動的な飛行を実現する。複数ではなく単機のドローンで三人称視点から操縦し、必要な機材の数を減らす。飛行する場所を 3D 再構築ではなく拡張現実の空間で認識し、GPS もマーカーも使用しないことで、準備時間の短縮とあらゆる場所での自動的な飛行を可能にする。GUI(Graphical User Interface)を数回タッチする容易な操作で、ドローンを自動的に飛行させる。

第3章 提案手法

本章では、まず本研究で使用する機材とコンピュータソフトウェアについて述べ、次にシステムの概要を示す。

3.1 使用する機材とコンピュータソフトウェア

3.1.1 ドローン

ドローンはRyze Tech社製TELLO IRON MAN EDITION(図7)を使用する。機体の重量は87g、サイズは98×92.5×41mmである。開発ツールとしてTELLO SDK(software development kit)[11]が提供されている。ドローンと操作端末をプログラムによるWiFi通信で接続し、既定のコマンドを送信してドローンの飛行を制御する。

図 7　Ryze Tech 社製 TELLO IRON MAN EDITION

3.1.2 拡張現実

拡張現実の構築には Apple 社が提供するフレームワークの ARKit[12]を使用し、Apple 社製の ios デバイス(スマートフォン iPhone やタブレット型コンピュータ iPad)[13]のカメラ映像を通して視認する。ARKit はカメラ映像の画像を解析した後、仮想のオブジェクトを映像に追加する。ios デバイスに備わっているジャイロセンサーや加速度センサーと連動させることで、あたかも現実空間に存在しているかのように見せかける(図 8)。ARkit を使用した場合、拡張現実の空間における座標の原点は ios デバイスで起動した地点となる。

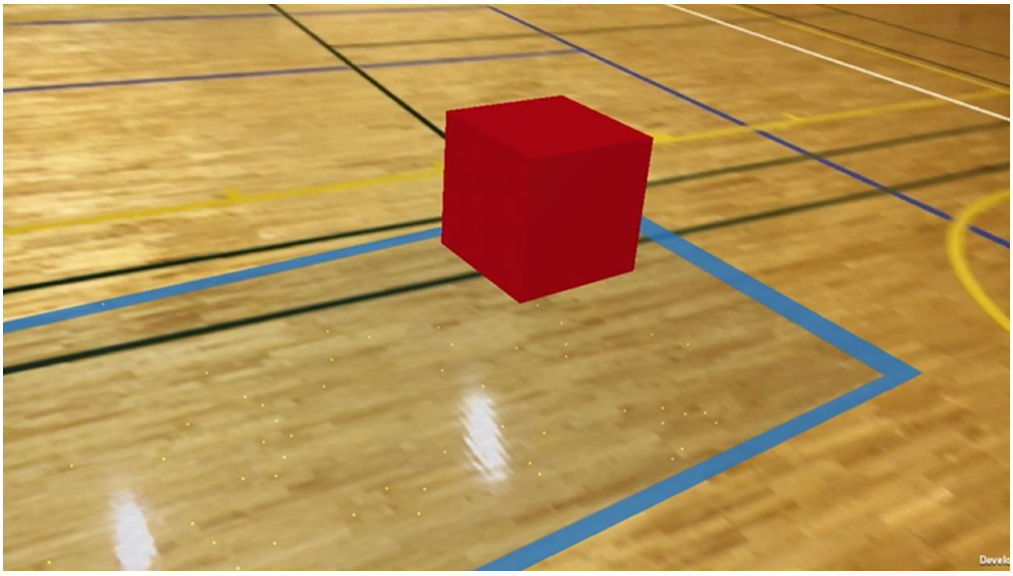

図 8 ARKit により 3D オブジェクトを現実空間へ配置する様子

3.1.3 モデリングとアニメーション

拡張現実空間に表示する 3D モデルとそのアニメーションの製作は、コンピュータグラフィックスソフトの Autodesk 社製 Maya[14]を使用した(図 9)。3D オブジェクトの形状を定義するポリゴンメッシュを作成する。3D オブジェクトの関節となるリグを設定する。設定されたリグを動かし 3D オブジェクトにアニメーションをつける作業をする。

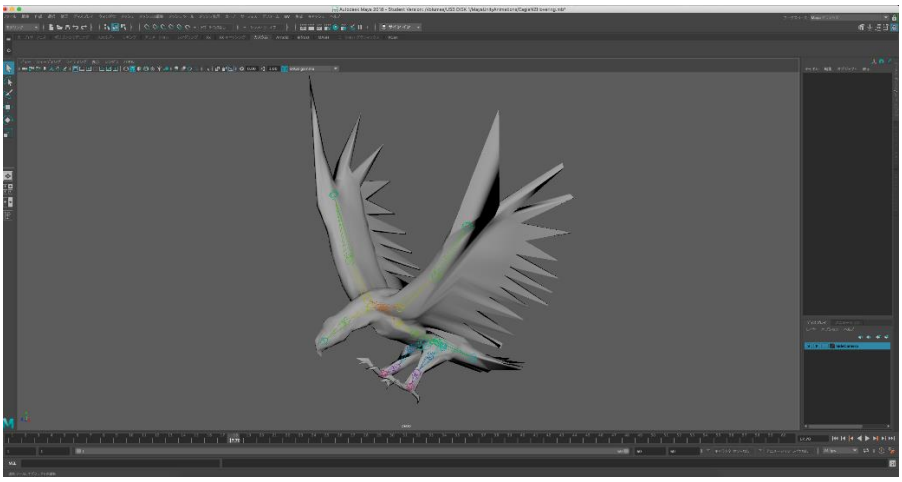

図 9 Autodesk 社製 MAYA

3.1.4 統合開発環境

ドローンの操縦、拡張現実の構築、GUI の表示を連携させるシステムの製作はゲームエンジンの Unity Technologies 社製 Unity[15]を使用した(図 10)。拡張現実の空間へのオブジェクト配置と当たり判定を実装する。ドローンと操縦者が操作するデバイスとの通信および制御、GUI の機能を設定する。

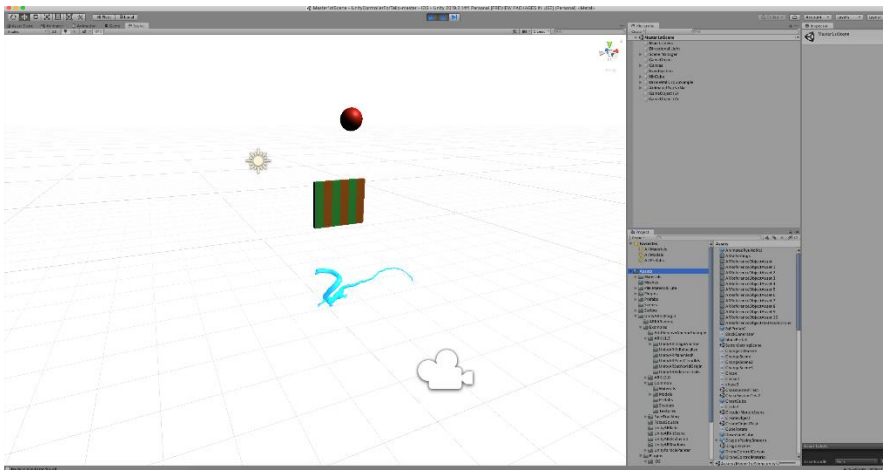

図 10 Unity Technologies 社製 Unity

3.2 システムの構成

システムの構成を図 11 に示す。

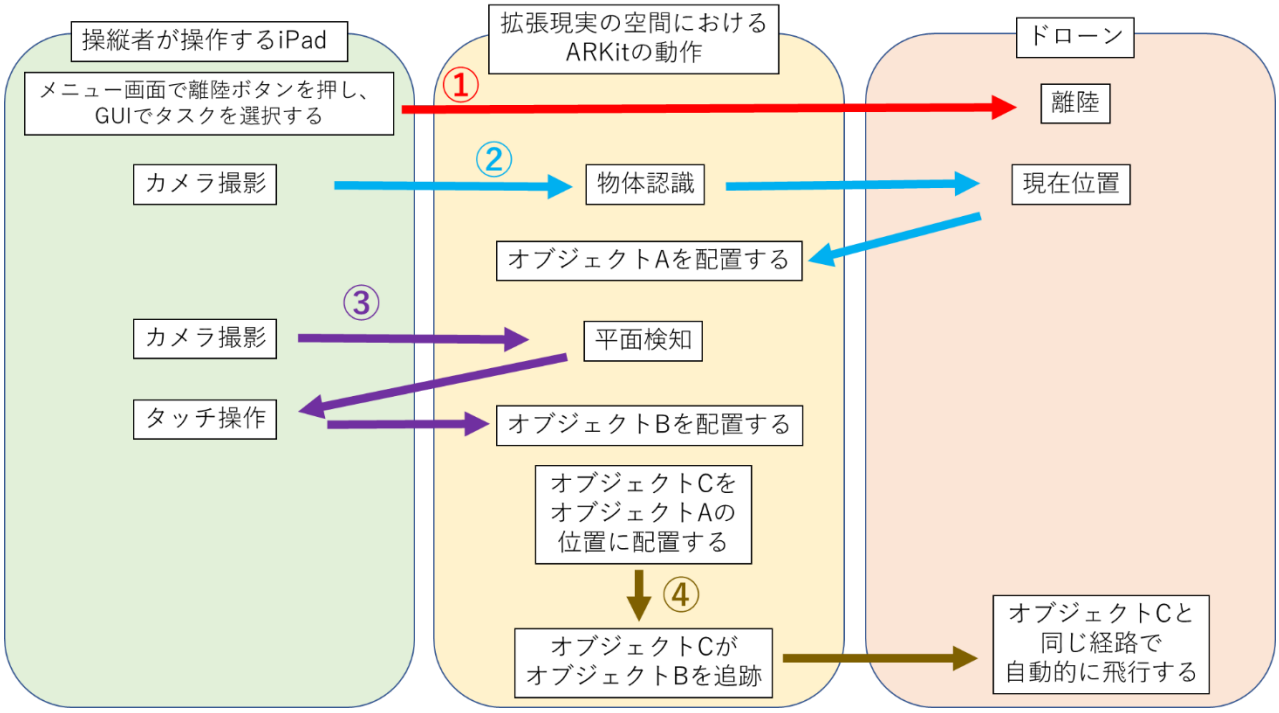

図 11 システムの概要

①操縦者が操作する iPad に表示されるメニュー画面で離陸ボタンを押し、GUI でタスクを選択する。(5.1 で解説する)

②iPad のカメラでドローンを ARKit によって物体認識し、拡張現実の空間でドローンと同じ位置にオブジェクト A を配置する。(4.1 で解説する)

③iPad のカメラでドローンが飛行する場所を ARKit によって平面検知し、検知した地点を画面上でタッチすることで拡張現実の空間へオブジェクト B を配置する。(4.2 で解説する)

④拡張現実の空間へオブジェクト B が配置されると同時に、オブジェクト A の地点へオブジェクト C が配置されオブジェクト B を追跡する。ドローンはオブジェクト C と同じ経路で自動的に飛行する。(4.3 で解説する)

第4章 システムの構築

本章では、本研究の提案手法を実現するためのシステムを構築する過程について述べる。

4.1 物体認識

4.1.1 物体のスキャン

ドローンを物体認識するためには、機体形状を撮影して割り出す特徴点のデータが必要である。データは ARKit が持つ機能の一つである ARKit Scanner[16]により作成する。まず iPad のカメラを物体へ向ける。画面には黄色い立方体の枠が表示されおり、枠をタッチ操作で引き延ばし物体を囲む。その後、物体を全周囲から撮影すると、黄色い点で表示されている特徴点を記録する。記録された特徴点のデータによって、自動的に拡張現実の空間で認識した位置を表示するアプリケーションである(図 12)。物体認識に必要な特徴点の数は特に指定されていないが、枠の中に表示される黄色い点の数が多いほど認識精度は向上する。

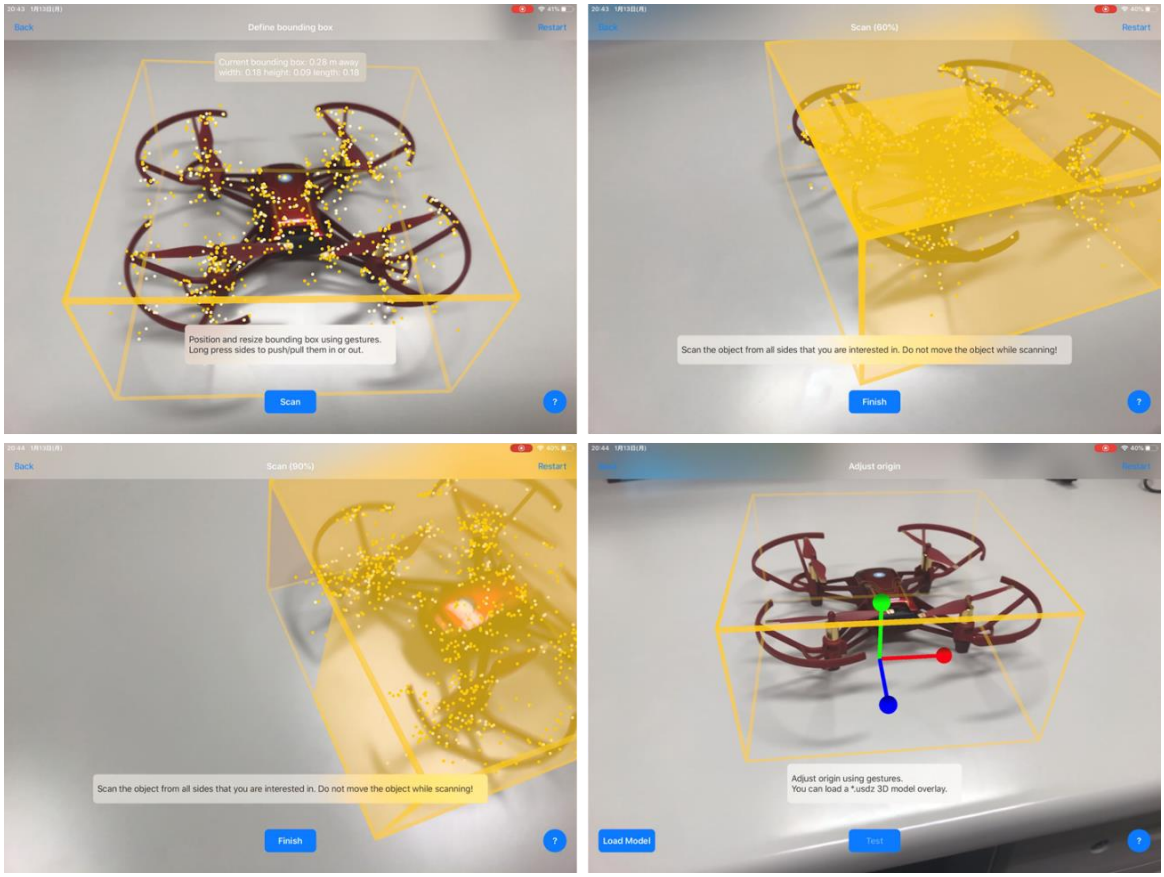

図 12 ARKit Scanner で特徴点を検出する

4.1.2 物体認識完了状態の表示

ARKit Scanner で検出した特徴点の情報を Unity へインポートすると、毎回スキャンする手間を省き物体認識をすることができる。iPad のカメラをドローンに向けるだけで、拡張現実の空間でドローンの位置と同じ位置へオレンジ色の球体であるオブジェクト A を生成し、ドローンが物体認識されたと一目でわかるようにした(図 13)。

図 13 ドローンを物体認識した位置にオブジェクト A を表示する。
(黄色い点は ARKit がカメラ映像から認識した特徴点)

4.2 3D オブジェクトの製作

本項目ではドローンの操縦者が飛行経路として拡張現実の空間へ配置するオブジェクト B の例について述べる。

まず、本研究では一例として円状に移動する球体のオブジェクトを制作する。円状の飛行は警備や農薬散布を想定した動きである。ドローンは警備業の深刻な人手不足を解消する機器として期待されている[17]。また、同じく農業の人手不足と労働負担の軽減目的とした農業用ドローン普及拡大に向けた方策が検討されている[18]。

次に、壁面を模した直方体のオブジェクトを制作する。壁面に沿った飛行は、建築物の点検を想定した動きである。ドローンは橋梁やトンネルの点検へ活用し、作業工数を縮減することが期待されている[19]。

最後に不規則な形状と動きを持つオブジェクトを制作する。実写映像の動きと 3D オブジェクトの動きを合成させることを想定したものである。

従来の合成する手法としてマッチムーブ[20]がある。この手法では実写の映像からカメラの移動量を割り出し、3D オブジェクト合成する。カメラの動きに合わせるので、3D オブジェクトの動きをクリエイターが自由に設定できない。

本研究の提案手法では、既に完成された 3D オブジェクトの動きに合わせてドローンが飛行することを目指すので、カメラの動きによって 3D オブジェクトの動きが制約されないようにした。

4.2.1 円状に移動する球体のオブジェクト

球体のオブジェクトは Unity の 3D オブジェクトを作成する機能で制作する。Y 軸を軸として回転する立方体の 3D オブジェクトを座標(0,0,0)の地点へ別に作成する。そのオブジェクトの動きと連動させることで、球体のオブジェクトは円状に移動する(図 14)。

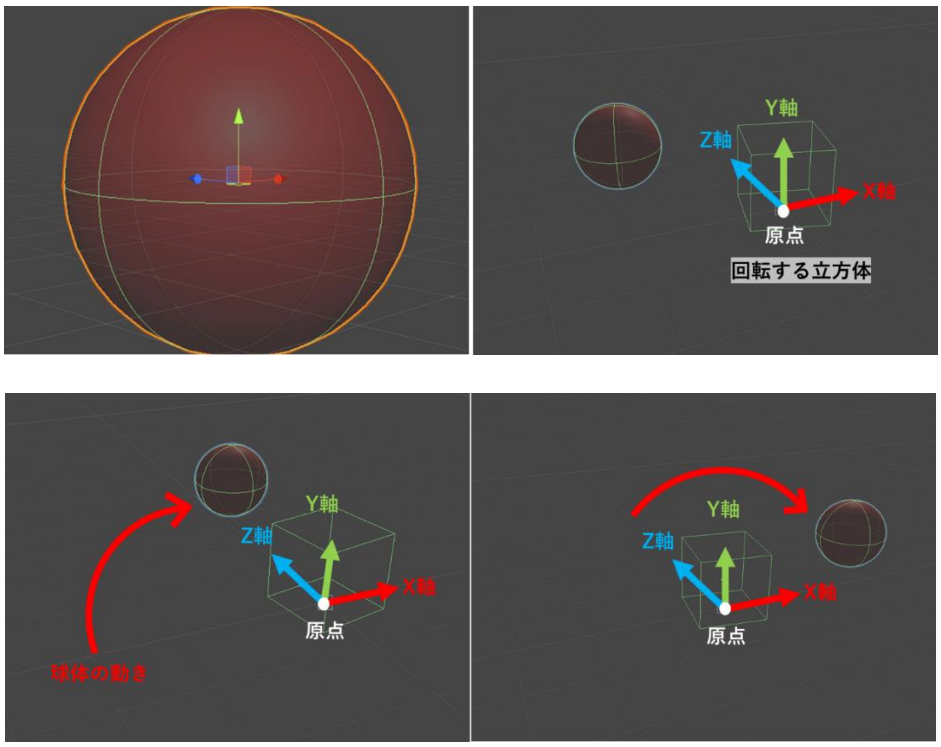

図 14 円状に移動する球体のオブジェクト

回転する立方体との距離を調節することによって、円状に移動する球体の旋回半径や高さを編集する(図 15)。Unity 上では 1 単位 1m であることを念頭に置き、拡張現実の空間で配置する際の旋回半径を X 座標で、飛行高度を Y 座標で入力あるいはカーソルで調整する。

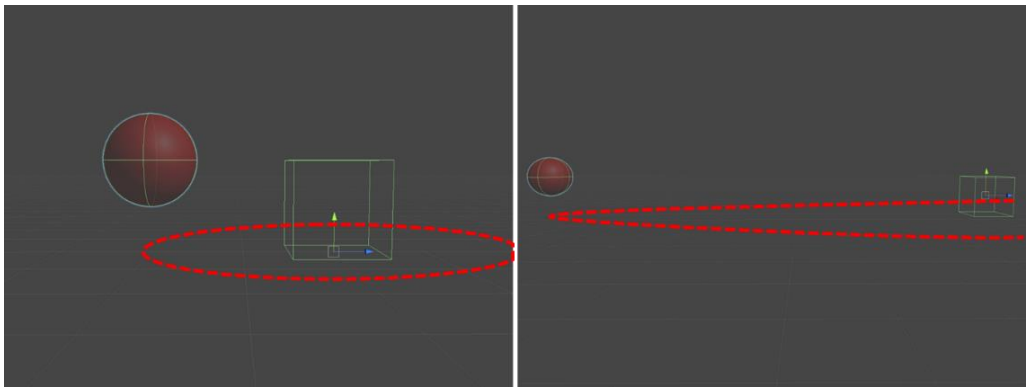

図 15 オブジェクトの調節

4.2.2 壁面を模した直方体

壁面のオブジェクトも、Unity の 3D オブジェクトを作成する機能で制作する。拡張現実の空間で GUI としての視認性を向上させる為に、1m 間隔で緑と赤の縞模様にした(図 16)。壁面の高さを任意に調節し、飛行させる環境に合わせて表示させることが可能である。

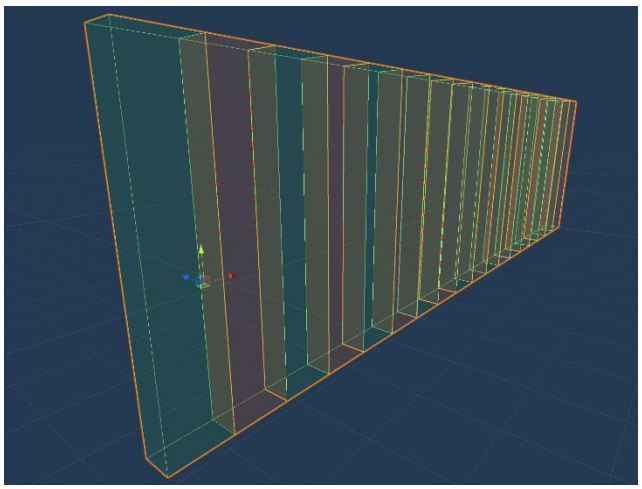

図 16 壁面のオブジェクト

次に、壁面のオブジェクトに沿って上下運動しながら横方向へ移動する球体のオブジェクトを作成する(図 17)。ドローンが動きを模すオブジェクト C には、この球体を追跡させるとする。動きはスクリプトで制御しており、上下する Y 軸の幅、スピードを入力することで任意の高さと速さで移動させる。

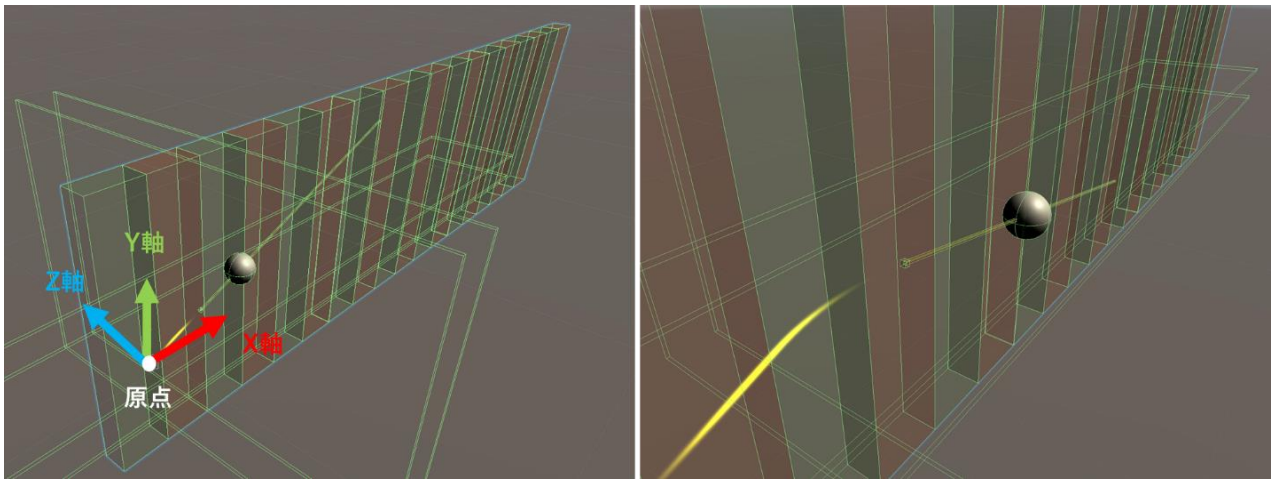

図 17 上下運動しながら横方向へ移動する球体

4.2.3 不規則な形状と動きを持つオブジェクト

複雑な形状と不規則な動きは Maya で制作する。形状をポリゴンメッシュで表現し、関節を設定するリギングを施し、ポーズを調節してアニメーションをつけた後、Unity へインポートする(図 18)。

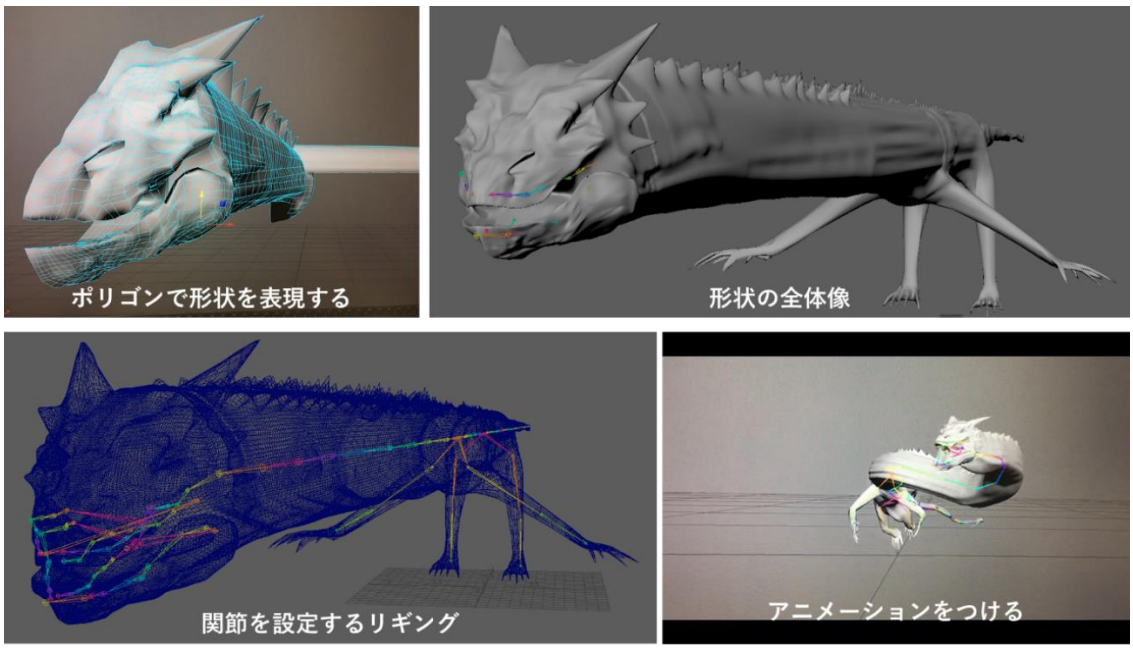

図 18 オブジェクトの製作過程

本研究ではしばらく操縦者がタッチした場所で揺らめいた後、回転しながら上昇するオブジェクトの挙動を使用する(図 19)。

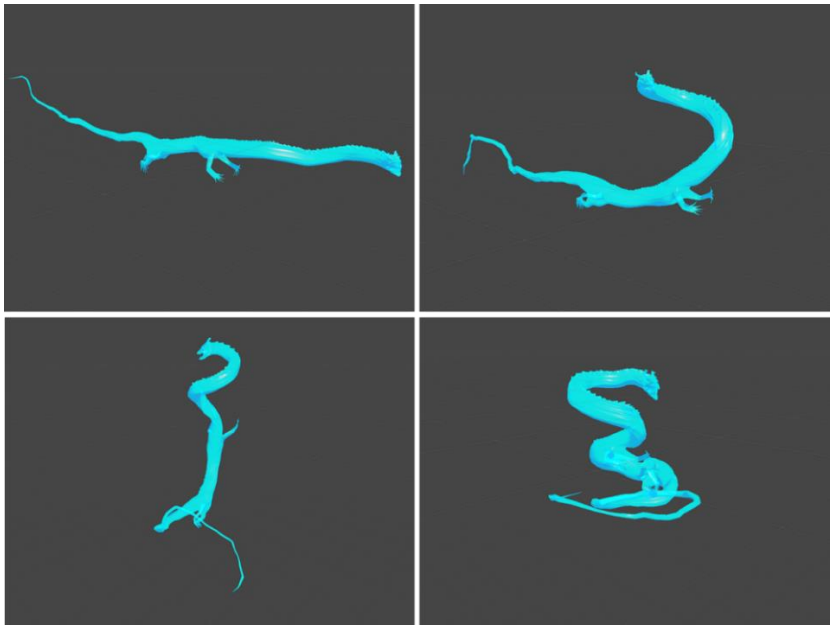

図 19 オブジェクトの動き

4.2.4 オブジェクトの配置

用意したオブジェクト B は拡張現実の空間に、ドローンの操縦者が配置する。配置する場所は iPad のカメラを通して ARKit が地面の特徴点を検出し、平面検知した範囲として表示される水色の枠線内から一か所を選択する。iPad で表示されている枠線内の場所をタッチして配置を完了する(図 20)。配置するオブジェクトの大きさや、タッチした地点からの高さは Unity で調整する。

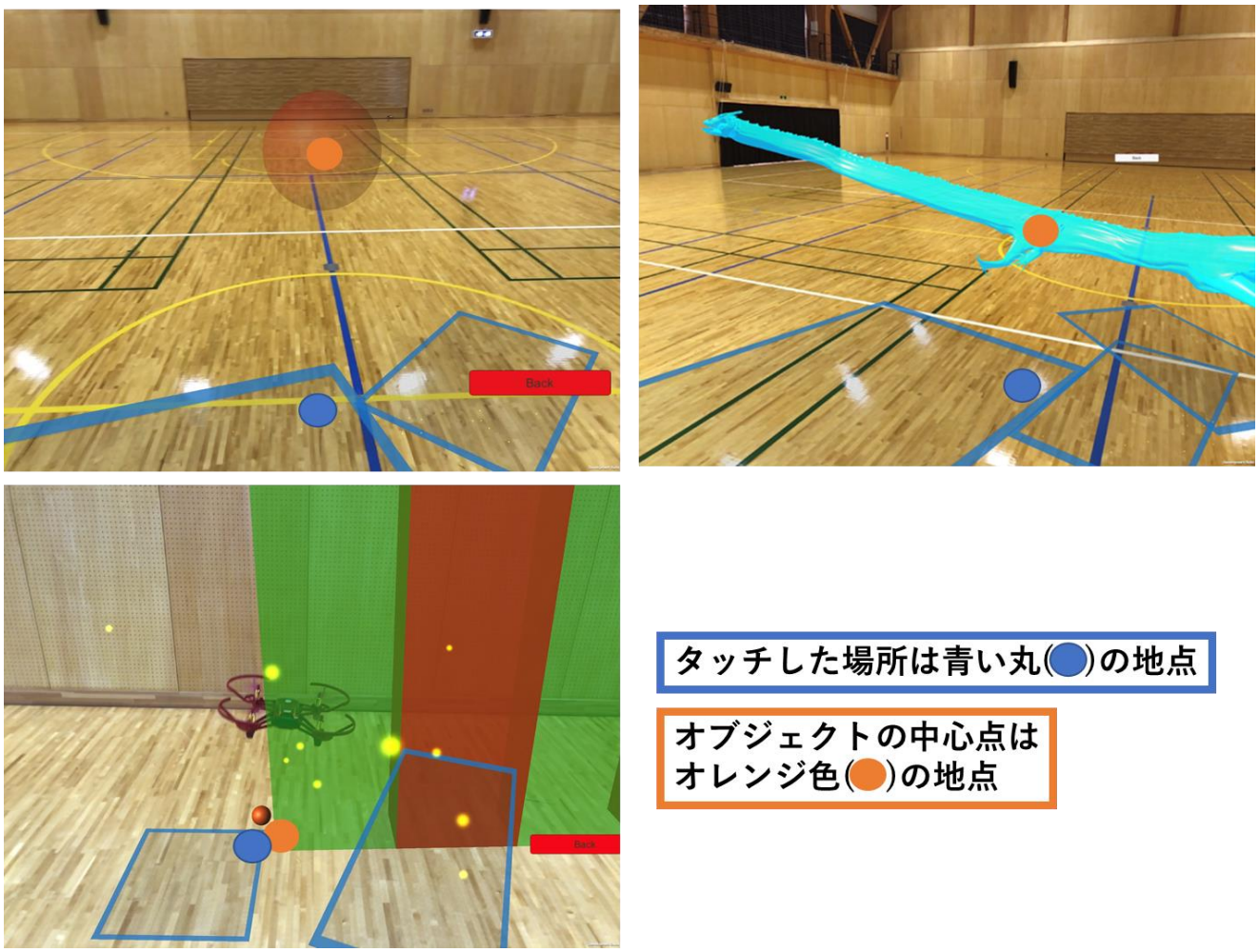

図 20 オブジェクト B は平面検知した場所の内一か所をタッチして配置する
(黄色い点は ARKit がカメラ映像から認識した特徴点)

4.3 ドローンの自動的な飛行

本項目では拡張現実の空間と連携したドローンの自動的な飛行について述べる。

4.3.1 追跡するオブジェクト

オブジェクト B はタッチ操作により拡張現実の空間へ配置されると、同時に生成されるオブジェクト C に追跡される。

オブジェクト C はドローンを物体認識した位置に生成されたオブジェクト A の位置から、オブジェクト B に向かってドローンが飛行する速度と同じ秒速 1.5m で移動する。また、オブジェクト C があまりにも急旋回するとドローンが動きに対応できないので、ドローンの性能に合わせて 2π/7 秒の角速度で回転するスピードとした(図 21)。

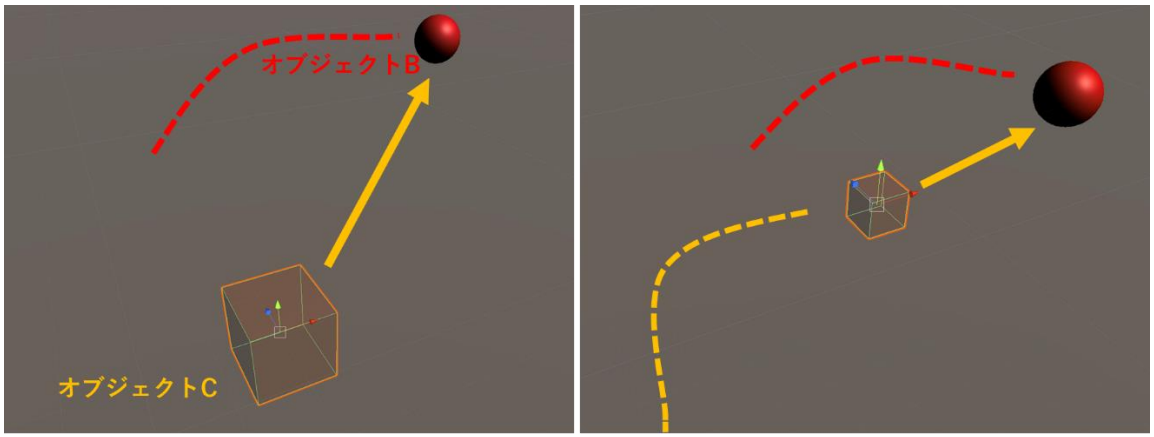

図 21 オブジェクト C に追跡されるオブジェクト B

4.3.2 当たり判定

オブジェクト C が拡張現実の空間で進行方向を変えると、ドローンも同じ進行方向へ自動的な飛行をするようにコマンドを iPad から送信するシステムを制作した。

図 22 を用いて解説する。

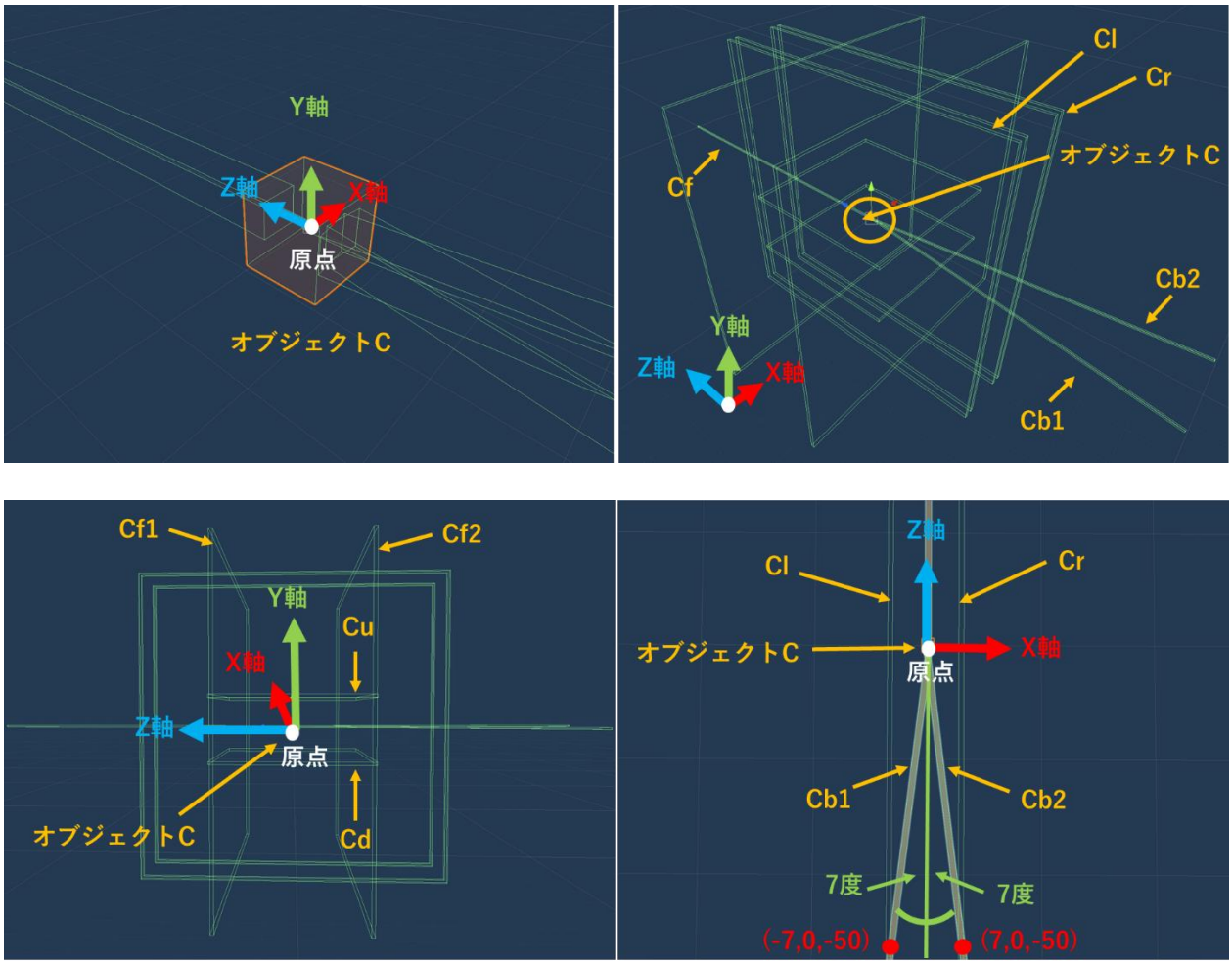

図 22 オブジェクト C と随伴するオブジェクト群

オブジェクト C から棒状のオブジェクトを正の Z 軸方向へ一本(Cf)、負の Z 軸方向へは Y 軸を軸にそれぞれ逆方向へ傾かせて二本(Cb1,Cb2)突き出す。傾きはオブジェクト C が 180 度回転した際、正の Z 軸側を正面として左右へ配置されている板状のオブジェクト(Cr,Cl)にちょうど接触する角度とする。また、板状のオブジェクトを上下(Cu,Cd)、前後(Cf1,Cf2)、左右(Cl,Cr)へ取り囲むように配置し回転させず随伴させる。これは、オブジェクト Cf が進行方向によって、接

触する板状のオブジェクトとの当たり判定が発生するごとに、ドローンへ同じ方向に飛行させるコマンドを WiFi 経由で送信することを目的としている。しかし、この方法ではオブジェクト C が 180 度回転した際、板状のオブジェクト Cr と Cl の当たり判定が現実世界で飛行しているドローンの向きと逆になってしまう。そこで 180 度回転して棒状のオブジェクト(Cb1,Cb2)が Cr と Cl の板に接触すると、右と左の判定を逆とする対策をしている。

4.3.3 コマンド送信

コマンドの一覧と拡張現実の空間で検知した当たり判定を検知によって、ドローンを飛行させる過程を図 23 に示す。

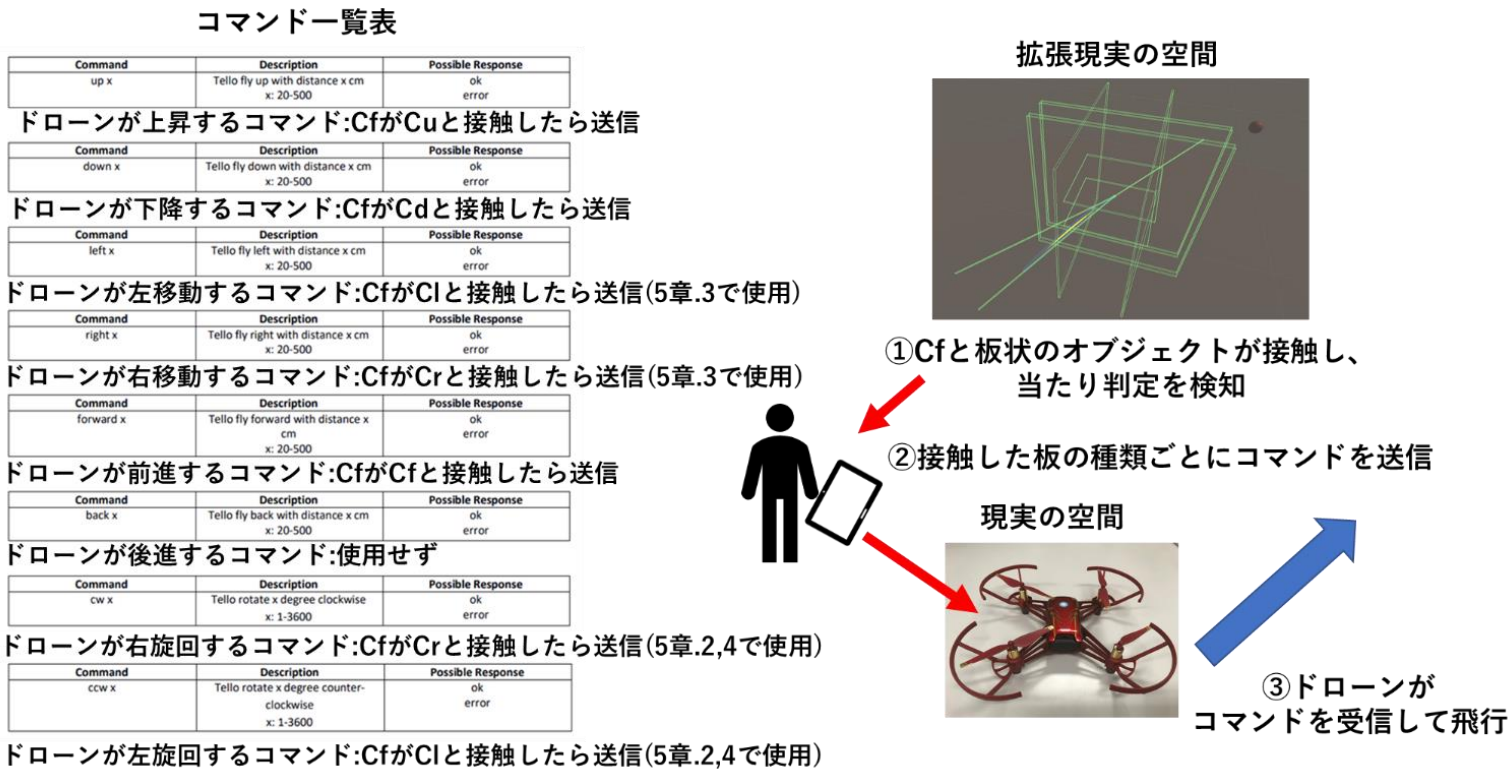

図 23 ドローンのコマンド(TELLO SDK[11])

4.4 複数方向への同時操縦

オブジェクト C に随伴する棒状のオブジェクト Cf が接触する板状のオブジェクトによって、ドローンへ送信されるコマンドの種類が決まるので図 24 のように複数の板と同時に接触した場合は、複数方向へのコマンドが同時に送信される。

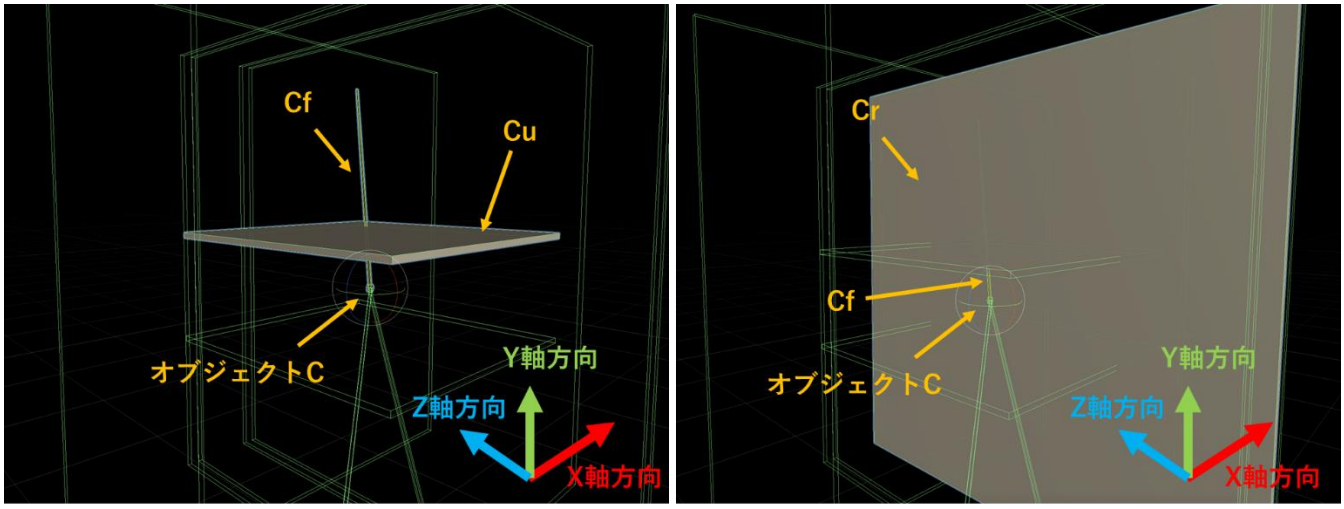

図 24　Cf が 2 枚の板(Cu,Cr)と同時に接触している

この場合、棒状のオブジェクト Cf は上側の板 Cu と接触しているのでドローンが上昇するコマンドが送信される。また、右側の板 Cr とも接触しているのでドローンが右旋回するコマンドが送信される。2 つのコマンドを受信したドローンは上昇しながら右旋回をする。

ジョイスティックを使用した手動による操縦では難しい、複数方向への同時操縦にも本研究の提案手法は対応している。

4.4.1 この章のまとめ

オブジェクト C は随伴するオブジェクト群と板状のオブジェクトとの当たり判定を検知しながら、オブジェクト B を追跡することでドローンの自動的な飛行を実現している(図 25)。

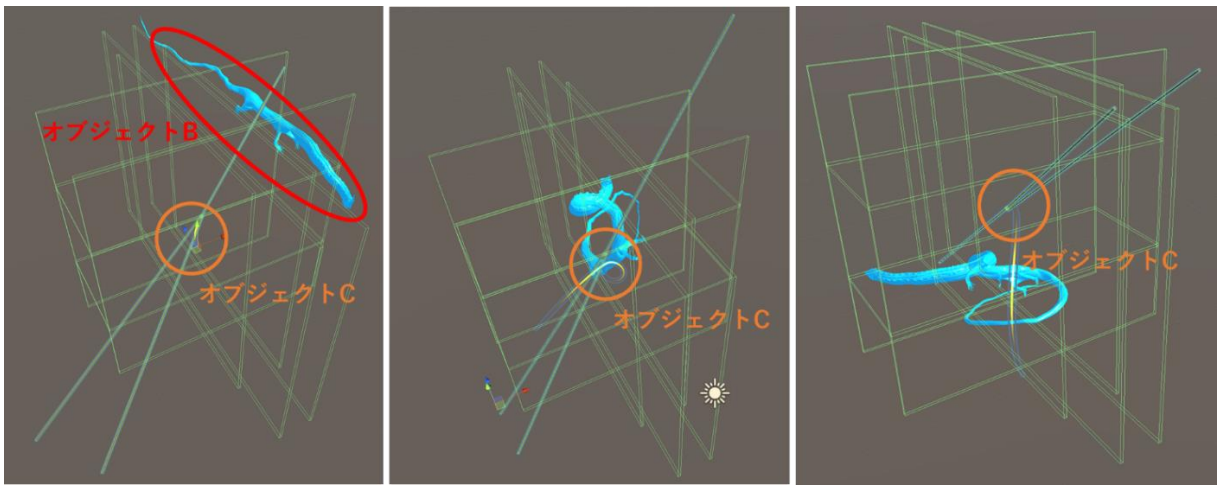

図 25 オブジェクト B を追跡するオブジェクト C と随伴するオブジェクト群

第5章 システム構築

本章では、構築したシステムを紹介する。まず、GUI について述べ、つぎにドローンに円運動をさせるモード、壁面に沿って飛行するモード、最後に複雑な動きに合わせて飛行するモードについて説明する。

5.1 GUI

ドローンに飛行させるモードを選択する画面を制作する。画面左側にはドローンを離陸させる”TakeOff”ボタンと着陸させる”Land”ボタンを作成した。

画面右側へ円状に飛行するモードへの遷移を”Circular Motion”と、壁面に沿って飛行するモードへの遷移を”Wall Inspection”と、複雑な動きに合わせて飛行するモードへの遷移を”Moving 3D Model”と表示するボタンを作成した。

ドローンに飛行させたい経路を文字で説明するだけではなく、ボタン付近にそれぞれのモードで表示するオブジェクトを配置する。文字とオブジェクトの両方を操縦者は確認することで、押すべきボタンが一目瞭然な GUI とする(図26)。

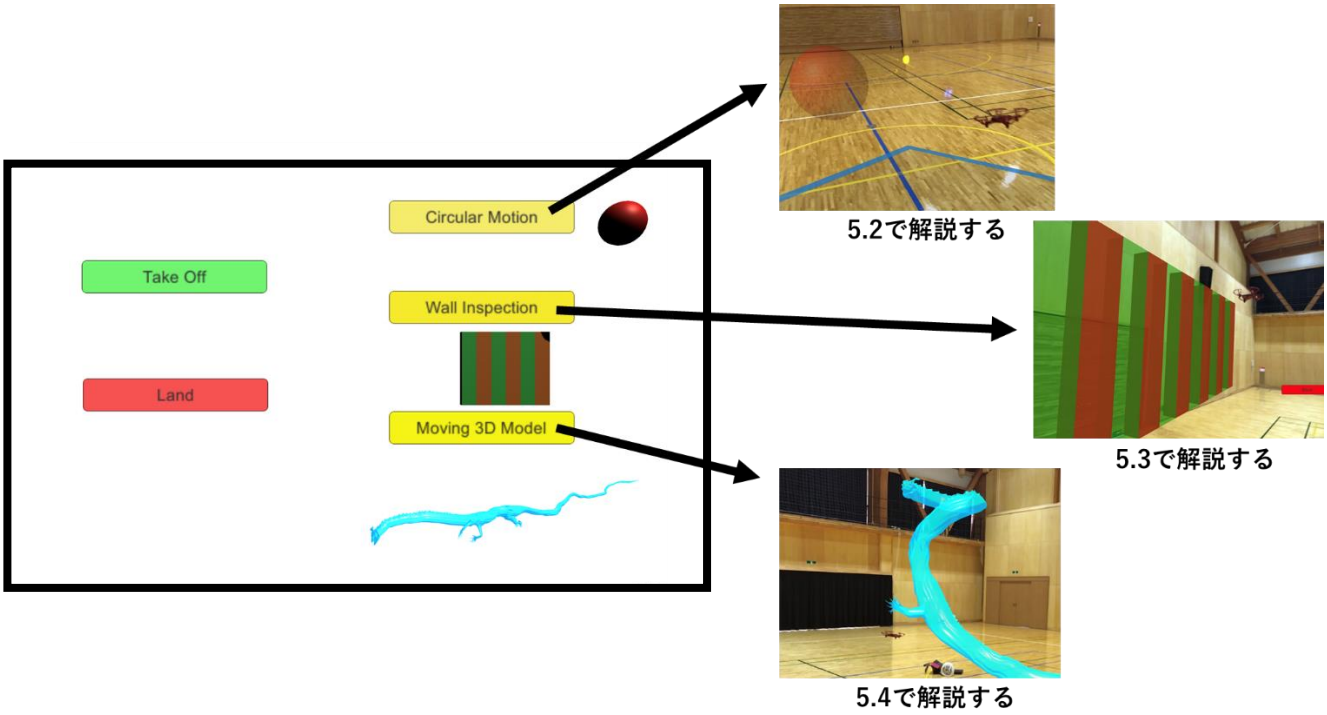

図 26　GUI

5.2 円状に飛行するモード

Circular Motion のボタンをメニュー画面で Take Off ボタンを押した後に選択する。iPad のカメラが起動したら、離陸しているドローンを物体認識してオブジェクト A を生成させる。生成を確認したら円状に飛行させたい場所で平面検知し、青い枠が表示された場所をタッチする。タッチした場所に円状に移動する赤い球体のオブジェクト B が拡張現実の空間に生成され、ドローンがそのオブジェクトと同じ動きで自動的に飛行する(図 27)。移動半径や生成する高さは Unity で予め調節する。

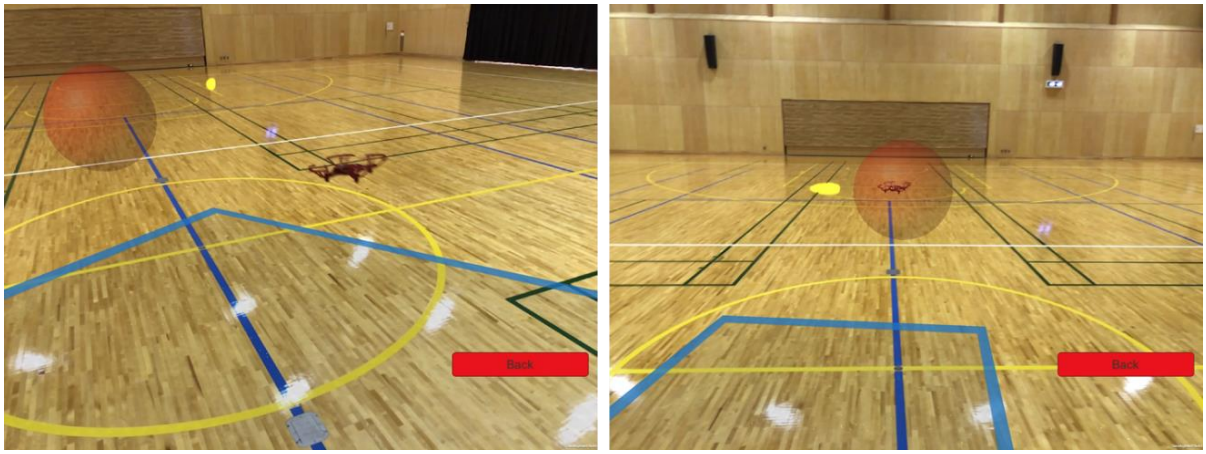

図 27 自動的に円状に飛行するドローン

(黄色い点は ARKit がカメラ映像から認識した特徴点)

5.3 壁面に沿って飛行するモード

メニュー画面で Take Off ボタンを押した後に Wall Inspection のボタンを選択する。iPad のカメラが起動したら、離陸しているドローンを物体認識してオブジェクト A を生成させる。生成を確認したらドローンを飛行させたい壁面の近くで平面検知し、青い枠が表示された場所をタッチする。タッチした場所に壁を模したオブジェクト B が拡張現実の空間に生成され、ドローンがそのオブジェクトの前で上下運動しながら横方向へ自動的に飛行する(図 28)。上下運動する幅や移動の速度は Unity で予め調整する。

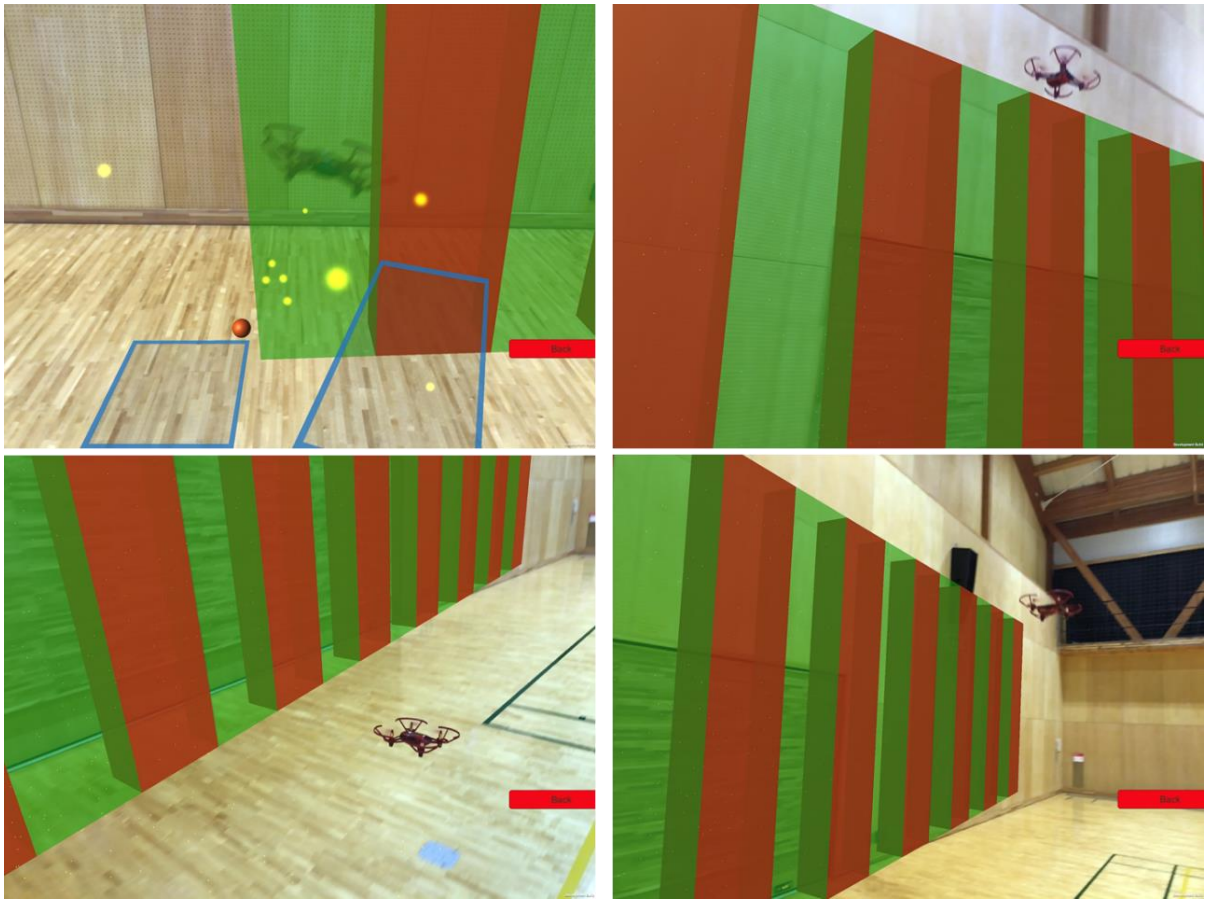

図 28 壁面に沿って飛行するドローン

(黄色い点は ARKit がカメラ映像から認識した特徴点)

5.4 複雑な動きに合わせて飛行するモード

メニュー画面で Take Off ボタンを押した後に Moving 3D Model のボタンを選択する。iPad のカメラが起動したら、離陸しているドローンを物体認識してオブジェクト A を生成させる。生成を確認したらオブジェクトを出現させたい場所で平面検知し、青い枠が表示された場所をタッチする。タッチした場所に不規則に動くオブジェクト B が拡張現実の空間に生成され、ドローンがそのオブジェクトと同じ動きで自動的に飛行する(図 29)。

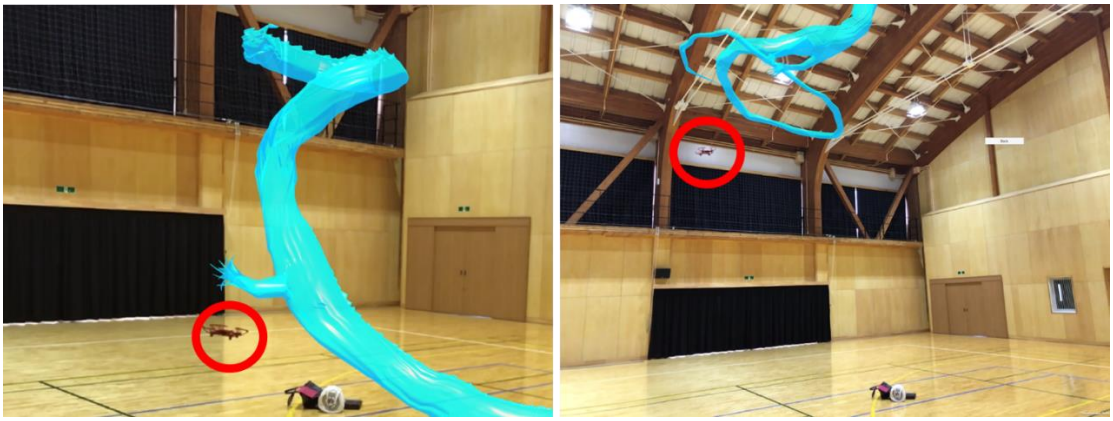

図 29 不規則な動きに合わせて飛行するドローン

操縦者はオブジェクト C を追跡させたい場所を、オブジェクトを制作する際に設定した関節から自由に Unity 上で設定する(図 30)。本研究では龍を模したオブジェクトを配置している。頭部を追跡させたい場合は頭部の関節を、胸部を追跡させたい場合は胸部の関節を選択することで、異なった部位の動きに応じた経路を飛行する。

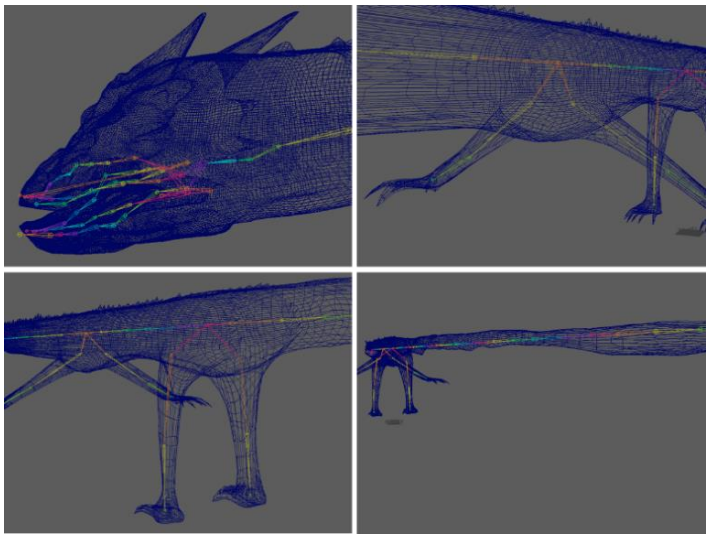

図 30 オブジェクトに設定されている関節

第6章 実験・評価

本章では提案手法を実験し評価する。まず実験の内容について述べ、次に計測したタイム、被験者へのアンケート結果をまとめる。

6.1 実験の環境

表 1 実験環境

被験者	本大学院の学生 10 名	
実験機材	ドローン	TELLO IRON MAN EDITION
	タブレット型コンピュータ	iPad Pro
実験場所	北陸先端科学技術大学院大学　体育館(図 31)	

図 31 実験場所

6.2 実験の目的

実験では被験者に、ドローンを本研究の提案手法とジョイスティックによる手動での操縦で飛行させる。同一の経路を飛行させて、その様子を動画撮影した映像からタイムを計測、比較し、アンケート調査を実施することで提案手法の有効性を考察する。

6.3 実験の内容

6.3.1 実験 1

実験 1 はドローンを円状に飛行させる。飛行経路はバスケットコートに描かれた半径 1.8m の円(図 32)を目安とし、カメラを進行方向へ向けつつ 3 周する。

図 32 バスケットコート

被験者は一回目の飛行で、ジョイスティックを使用した手動による操縦をしてタイムを計測する。二回目の飛行は提案手法でドローンを自動的に飛行させる(図 33)。

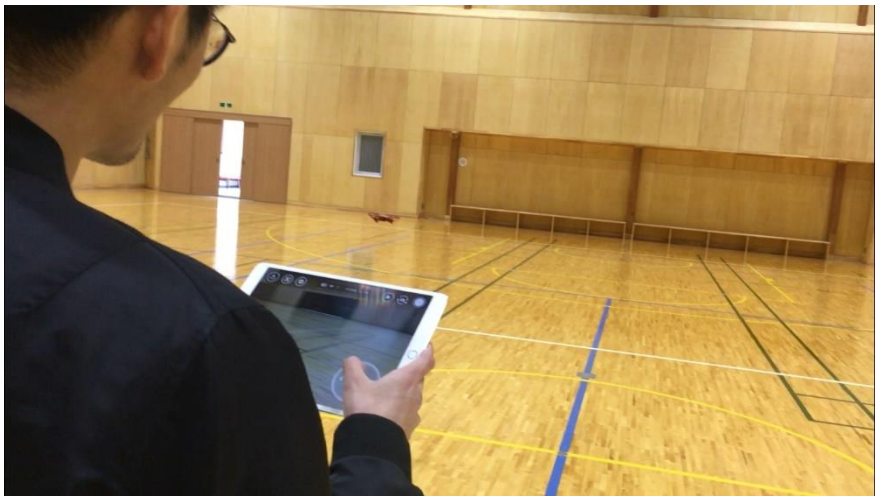

図 33　実験 1 の様子

6.3.2 実験 2

実験 2 はドローンを体育館の壁面に沿って上下運動させながら操縦者から見て右方向へ飛行させる。上下運動の範囲は壁に走っている二本の線(図 34 の赤線)、右方向への移動は 15m とする。

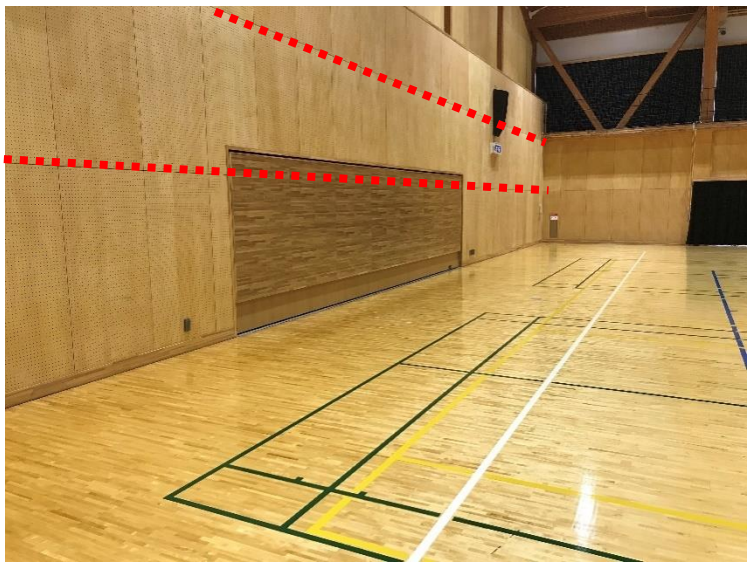

図 34 体育館の壁面

被験者は一回目の飛行で、ジョイスティックを使用した手動による操縦をしてタイムを計測する。二回目の飛行は提案手法でドローンを自動的に飛行させる(図 35)。

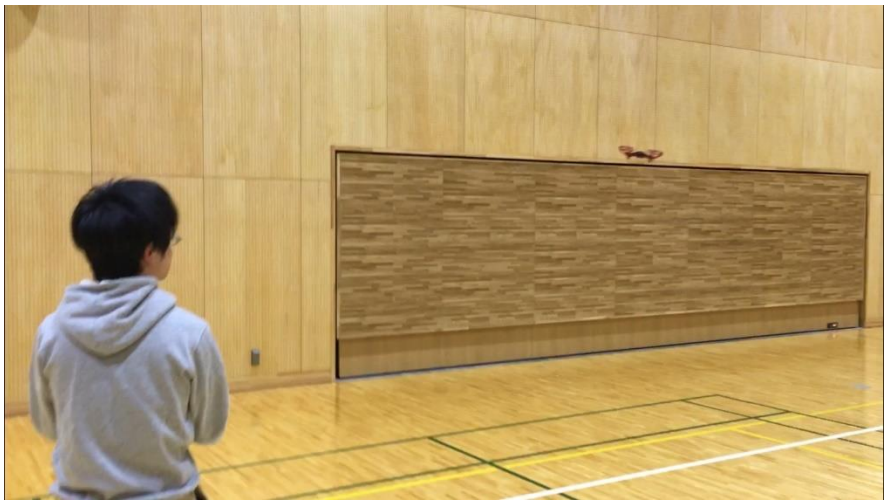

図 35 実験 2 の様子

6.4 計測したタイム

6.4.1 実験 1 で計測したタイム

表 2 タイム(単位:秒)

※被験者が実験途中で操縦の続行が不可能と判断した場合、断念と表記

※システムが正常に動作しなかった場合、失敗(回数)と表記する。

	手動での操縦	本研究の提案手法
被験者 1	47	28
被験者 2	53	28 失敗(1)
被験者 3	30	27
被験者 4	断念	27 失敗(2)
被験者 5	37	27
被験者 6	断念	28 失敗(1)
被験者 7	30	29 失敗(1)
被験者 8	64	27 失敗(1)
被験者 9	65	28 失敗(2)
被験者 10	50	27
平均	47.0	27.7

6.4.2 実験 2 で計測したタイム

表 3 タイム(単位:秒)

※被験者が実験途中で操縦の続行が不可能と判断した場合、断念と表記

※システムが正常に動作しなかった場合、失敗(回数)と表記する。

	手動での操縦	本研究の提案手法
被験者 1	20	7
被験者 2	65	9 失敗(1)
被験者 3	17	8
被験者 4	82	7
被験者 5	90	8
被験者 6	84	7
被験者 7	90	8
被験者 8	74	7
被験者 9	断念	8
被験者 10	95	7
平均	68.6	7.6

6.5 アンケート調査

実験の終了後、被験者にアンケート調査を実施した(図 36)。

本研究の提案手法に関するアンケート

氏名：

① 手動によるドローンの操縦経験の有無、頻度について記入してください。

② 手動でドローンを操縦する際、難しいと感じる点を記入してください。

③ 本研究の提案手法によるドローンの操縦と手動による操縦を比較し、設問ごとに該当する５段階評価の場所へ○を記入してください。

（５とても容易である４容易である３どちらでもない２困難である１とても困難である）

1．使用方法を短時間で習得することは容易でしたか？

5　4　3　2　1

とても容易である　　とても困難である

2．自動化されたドローンの操縦は容易でしたか？

5　4　3　2　1

とても容易である　　とても困難である

3．飛行場所をタッチ操作によるオブジェクト配置で指定することは容易でしたか？

5　4　3　2　1

とても容易である　　とても困難である

4．飛行経路を AR 空間への表示によって把握することは容易でしたか？

5　4　3　2　1

とても容易である　　とても困難である

5．目的に合わせた操縦を GUI で選択することは容易でしたか？

5　4　3　2　1

とても容易である　　とても困難である

図 36 本研究に関するアンケート

表 4 アンケート回答(設問①,②)

	設問①	設問②
被験者 1	経験有 半年に一回操縦	前進とせん回の組み合わせ
被験者 2	無	直線運動をさせたい場合、ドローンの位置によって入力する角度が変わる点
被験者 3	半年ほど前に数回	操作と実動作の誤差が一定でない。ラグ
被験者 4	有　10 回	ドローンの正面方向と操縦の方向が違う
被験者 5	2 年前仕事で	ドローンの正面方向と操縦の方向が違う
被験者 6	ない	回しにくい
被験者 7	経験無	ドローンのカメラの映像と、現実のドローンの位置関係
被験者 8	今まで一度もない	同時にカメラ操作と、ドローン操作をするのが難しかった。
被験者 9	無	ドローンの向きがわかりにくい
被験者 10	過去 2 回	スピードの調整

表 5 アンケート回答(設問③)

	設問③1	設問③2	設問③3	設問③4	設問③5
被験者 1	5	4	4	2	4
被験者 2	4	5	3	5	5
被験者 3	4	5	2	1	5
被験者 4	5	5	4	4	5
被験者 5	4	5	5	5	5
被験者 6	5	5	5	4	5
被験者 7	4	5	3	4	4
被験者 8	4	5	3	4	4
被験者 9	4	4	5	5	5
被験者 10	5	4	5	4	4

表 6 設問③における回答の平均値

	回答の平均値
設問③1	4.4
設問③2	4.7
設問③3	3.9
設問③4	3.8
設問③5	4.6

第7章 実験の結果と考察

本章ではまず評価実験の結果について述べ、次に考察する。

7.1 実験の結果

7.1.1 実験のタイムについて

本研究の提案手法による自動的な操縦の平均タイムは、手動による操縦での平均タイムと比較して実験 1 では 19 秒、実験 2 では 61 秒短かった。拡張現実の空間と連動したドローンの自動的な飛行は、指定した経路の飛行を効率化すると示している。

7.1.2 アンケート調査の結果について

設問①の回答で、被験者のドローンを操縦した経験について確認した。

設問②の回答から、ドローンの手動による操縦では方向を制御することが特に困難であるとわかった。その原因として操縦者の体の向きとドローンの向きが異なること、遠くからではドローンの視認が困難であることが指摘されている。その他、複数の方向への同時操作、スピードの調整、ジョイスティックの入力がドローンの飛行に反映されるまでに時間差があることを難しいと感じる被験者がいた。

設問③1 における回答の平均値は 4.4 で、提案手法でドローンを自動飛行させるまでの手順を把握することが、ジョイスティックの入力方向でドローンの飛行を制御させることと比較して容易であると示している。

設問③2 における回答の平均値は 4.7 で、提案手法により自動化されたドローンの操縦は、ジョイスティックを使用した手動によるドローンの操縦と比較して容易であると示している。

設問③3 における回答の平均値は 3.9 で、提案手法で拡張現実の空間へオブジェクト配置することにより飛行場所を指定することは、飛行場所までドローンを手動で移動させることよりも容易であると示している。

設問③4 における回答の平均値は 3.8 で、飛行経路を拡張現実の空間へ表示することは、ドローンを直接視認しながら行う飛行経路の確認と比較して容易であると示している。

設問③5 の平均値は 4.6 で、ドローンに目的と合致した経路を飛行させる場合、提案手法で制作したアプリケーションの GUI が、実験で使用したアプリケーションと比較して使用が容易であると示している。

7.2 考察

タイムの計測結果とアンケート調査を総合的に考察する。手動による操縦のタイムが長くなる原因は、アンケート調査の設問②で被験者が指摘した問題点と関係している。飛行しているドローンの方向が把握できないこと、スピードの調節や複数方向への同時入力が難しいこと、操作が飛行に反映されるまでに時間差があることは飛行経路からの逸脱につながる。また、逸脱した経路から正しい経路へ戻る際にも障害となり、タイムが伸びてしまう。アンケート調査の設問①とタイム結果から、ドローンの操縦経験の有無に関わらず適切な経路を飛行させることは困難であるとわかる。

2 つの手法のタイム差は実験 2 が実験 1 と比較して 41 秒大きい。実験 2 の飛行経路には実験 1 には含まれていない上下運動がある。よって、手動で操縦する場合ドローンを上下運動しつつ横方向へ飛行させることは、円状に飛行させること以上に困難であると推察する。操縦の自動化によって飛行時間を短縮することは、上下運動を伴った飛行経路でより効果を発揮すると期待できる。

手動による操縦の実験 1 で被験者 4 と被験者 6 が、実験 2 で被験者 9 がドローンを飛行経路から大きく逸脱させ、復帰させる操縦がわからず断念している。アンケート調査を確認すると、全員がドローンの方向を把握することが困難であると回答している。本研究の提案手法による自動的な操縦では、三名ともドローンを飛行させている。提案手法を用いることでドローンの手動による操縦が苦手な者でも、指定された経路に沿った飛行が可能であると示している。

手動による操縦では、被験者によってタイムに大きな差がある。実験 1 で最も短いタイムは被験者 3、7 の 30 秒で、最も長いタイムは被験者 9 の 65 秒で 35 秒のタイム差がある。実験 2 で最も短いタイムは被験者 3 の 17 秒で、最も長いタイムは被験者 10 の 95 秒で 78 秒のタイム差がある。一方、本研究の提案手法では実験 1、2 共に最短のタイムと最長のタイムの差が 2 秒以内となっており、操縦者によってタイムが大きく変わることがない。すなわち、個々人のスキルに差が出ず誰でも使えると言える。

第8章 課題と展望

本章ではまず本研究をまとめ、次に提案手法が抱える課題について説明し、最後に展望を述べる

8.1 まとめ

本研究では拡張現実の空間で移動するオブジェクトの経路を、ドローンの飛行経路とする手法を提案した。GPS が使用できない環境下や、マーカーの配置が困難な場所、あるいはコンピュータ上に 3D 再構築されていない空間においてもドローンを自動的に飛行させた。

拡張現実の空間をタブレット型コンピュータから視認しながら、タッチ操作でドローンを自動的に飛行させることで、手動による操縦が困難な人でも簡単に任意の経路を指定可能とした。

8.2 課題

本研究の提案手法で実験した際システムが正常に動作せず、拡張現実の空間でオブジェクトが移動している経路をドローンが認識しない場合があった。原因は、システムの処理落ちであると推察する。拡張現実の空間における当たり判定をタブレット型コンピュータで検知し、飛行させる方向を指示するコマンドをドローンへ通信しているが、当たり判定の処理はコンピュータに大きな負荷がかかる。その為、継続的な通信が阻害され、拡張現実の空間で移動するオブジェクトと現実の空間で飛行するドローンとの間で位置の差異が発生したと考えられる。

システムの処理落ちは拡張現実の空間に配置されたオブジェクトを、タブレット型コンピュータへ表示する際にも大きな障害となった。オブジェクトの配置はドローンが飛行する経路を操縦者へ表示する重要な要素であるが、処理落ちによってカメラ映像のフレームレートが低下し、表示を認識することが困難となった。被験者へのアンケート調査で表示に関する設問③4 の平均値が最も低い値であり、提案手法における重大な課題となっている。

拡張現実の空間でオブジェクトが動く範囲の変更は、予め Unity でプログラムを書き換えて設定する必要がある。システムをビルドし直すこと無く、動作させながら変更する機能を付け加えることで、利便性が向上すると推察する。

8.3 展望

本研究は拡張現実の空間におけるオブジェクトの動きのみから、ドローンを自動的に飛行させている。その為、現実空間に存在する障害物に対応していない。ドローンに搭載されているカメラからの映像を画像認識することや、赤外線センサーを活用することで障害物を自動的に回避する機能を実装し、飛行の安全性を向上させたい。また、位置ズレの修正への活用にも期待する。

構築したシステムは一機のドローンにのみ対応しているが、複数機体の操縦へ対応させることで編隊飛行を実現したい。編隊飛行によって広範囲を飛行することや、物体の撮影や点検に要する時間を短縮すること、指定した地点や物体を取り囲む様にドローンを配置させることが可能となる。

拡張現実の空間を複数人で共有し、飛行しているドローンの情報を多人数へ伝達したい。1 つのドローンを共同で使用することや、飛行経路を確認し合うことで集団における作業の効率化や安全性を向上させたい。

第9章 謝辞

本研究へご指導して頂きました宮田一乗教授、謝浩然助教授へ心より感謝いたします。研究方針が二転三転する中、技術的な課題、および実用性において様々なご指摘を受け、最終的にテーマを絞ることができました。また、関連研究や事例を多数ご紹介いただき、大変参考になりました。ゼミにおける先生方との質疑応答を通して、研究の目的や意義、新規性を客観的に捉えることの重要性を学びました。

実験にあたり、北陸先端科学技術大学院大学の関係者の方々に、体育館でドローンを飛行させた前例がないにも関わらず、使用を許可して頂きました。感謝いたします。

多忙の中、寒い環境下でドローンの操縦と本研究の提案手法に関する説明を受け、実験へ参加して頂いた被験者の方々に感謝いたします。

共に過ごした宮田研究室のみなさまに感謝を申し上げます。専門分野が異なる人の研究や知識に大変感銘を受けました。自身が明るくない分野における最新技術の動向も耳に入り、知見が広がりました。

最後に、交流を深めた留学生のみなさまに感謝いたします。故郷の話を伺ったことや、祖国の食べ物をお土産にいただいたことは得難き体験でした。研究においても独自の視点を持たれていて、興味深い内容でした。

参考文献

[1] 特許庁,平成 30 年度 特許出願技術動向調査　－ドローン－,
https://www.jpo.go.jp/resources/report/gidou-houkoku/tokkyo/document/index/30_05slide.pdf

[2] 下ノ村　和弘,ドローンの技術的側面 現状および課題と展望,IATSS Review,Vol. 44,No. 2,pp.16-23(2019)

[3] GPS 技術入門, 坂井　丈泰, 東京電機大学出版局(2003)

[4] Mohammad Fattahi Sani, Ghader Karimian, "Automatic Navigation and Landing of an Indoor AR. Drone Quadrotor Using ArUco Marker and Inertial Sensors",IEEE, 2017 International Conference on Computer and Drone Applications,pp102-107 (2017)

[5] 菊池　慶仁,加島 正爽,AR マーカーに基づくドローンの自立飛行,工学研究：北海学園大学大学院工学研究科紀要(18),pp.33-37(2018)

[6] Christoph Gebhardt, Benjamin Hepp, Tobias Naegeli, Stefan Stevsic, Otmar Hilliges," Airways: Optimization-Based Planning of Quadrotor Trajectories according to High-Level User Goals",
the 2016 CHI Conference on Human Factors in Computing Systems, pp1-12(2016)

[7] Ryotaro Temma,Kazuki Takashima,Kazuyuki Fujita,
Koh SuedaYoshifumi Kitamura,
"Third-Person Piloting: Increasing Situational Awareness using a Spatially Coupled Second Drone",
Human-Robot Interaction,pp507-519(2019)

[8] Okan Erat, Werner Alexander Isop, Denis Kalkofen
and Dieter Schmalstieg,
"Drone-Augmented Human Vision: Exocentric Control for Drones Exploring Hidden Areas",
IEEE Transactions on Visualization and Computer Graphics, pp1437-1446(2018)

[9] Ziquan Lan, Mohit Shridhar, David Hsu,Shengdong Zhao ,
"XPose: Reinventing User Interaction with Flying Cameras",
Robotics:Science&Systems(2017),9pages(2017)

[10] Benjamin Hepp,Matthias Nießner,Otmar Hilliges,
"Plan3D: Viewpoint and Trajectory Optimization for Aerial Multi-View Stereo Reconstruction",
ACM Transactions on Graphics,vol.38,no.1,pp4:1-4:17(2017)

[11] Rize Tech,TELLO ダウンロード,
https://www.ryzerobotics.com/jp/tello/downloads
[12] Apple,ARKit, https://developer.apple.com/jp/documentation/arkit/
[13] iOS,https://www.apple.com/jp/ios/ios-13/(2020)
[14] Autodesk,MAYA, https://www.autodesk.co.jp/products/maya/overview
[15] Unity Technologies,Unity,https://unity.com/ja
[16] Apple,ARKit Scanner,
https://developer.apple.com/documentation/arkit/
scanning_and_detecting_3d_objects
[17] 警察庁　人口減少時代における警備業務の在り方に関する有識者検討会,
人口減少時代における警備業務の在り方に関する報告書(2018)
[18] 農林水産省,
農業用ドローンの普及に向けて（農業用ドローン普及計画）(2019)
[19] 国土交通省 公共事業企画調整課,
次世代インフラ用ロボットの開発・導入について,
https://www.zenken.com/kensyuu/kousyuukai/H30/651/651_kondo.pdf
[20] CGWORLD,https://entry.cgworld.jp/terms/