

ドローンを用いたプログラミング教育のための教材の開発

福井 真二

情報教育講座

Development of Educational Material for Programming Education Using Drone

Shinji FUKUI

Department of Information Sciences, Aichi University of Education, Kariya 448-8542, Japan

1. はじめに

近年、プログラミング教育に対する関心が高まっている。小学校におけるプログラミング教育の実施が学習指導要領 [1] に明記され、令和2年度から小学校でプログラミング教育が行われるようになった。それに伴い、様々なプログラミング教育用教材が開発されている。また、プログラミングを教える塾や教室も増えてきている [2]。

プログラミング教育では、プログラミングの学習者（以下、学習者という）が興味を持てるような教材が必要とされている。特に、小学校でプログラミングに初めて触れるような児童にとっては、教材に対して興味を持てるかどうかということは非常に重要である。

学習者が興味を持ってプログラミングを学習できると思われる教材の1つに、ロボットを動作させるプログラムの作成を通してプログラミングを学習する教材がある。学習者が作成したプログラムによって目の前に存在するロボットを学習者の思い通りに動作させることができた時、学習者は喜びや楽しさを感じる。それらが動機となり、学習者がプログラミングを学び続けるようになることが期待される。このような理由から、ロボットを用いたプログラミング教育用教材がいくつも市販されている [3]。それらの教材の問題点として、多くの教材が1人1台のロボットを利用することを想定しており、学習環境を準備するのに費用がかかるというものがある。

筆者はこれまでロボットカー [4] を制御するプログラムを作成することを通してプログラミング教育を行う教材を開発してきた [5] [6]。この教材で使用されるロボットカーは、Raspberry Pi、モータードライバー及び市販されている安価な部品によって簡単に作成できる。かつ、無線LAN経由で動作を制御できるようになっており、1台のロボットカーを複数人で利用

できる。そのため、比較的安価にプログラミング学習環境を整えられる。また、この教材はScratch [7] によってロボットカー制御用プログラムを作成できるようになっている。Scratchは、ブロックを組み合わせてプログラムを作成できるビジュアルプログラミング環境である。Scratchを利用すればプログラムコードを入力することなくプログラムを作成可能であり、小学校でプログラミング教育を行う際にScratchはよく利用されている。この教材を使ってプログラミングを学習する際、ロボットカー制御用プログラムをScratchで作成するため、プログラムの作成方法が分からなくて学習者が困ることは少ない。これらの理由から、この教材は小学校でプログラミング教育を行う際に利用する教材として適しているといえる。

この教材の問題点として、手軽に利用できないということがあった。この教材を小学校で利用する場合、教員が予めロボットカーを作成しておく必要がある。ある程度電気回路に関する知識がある人であれば、文献 [4] [5] を参考にしながらロボットカーを組み立てられるが、電気回路に関する知識がない人にはロボットカーを組み立てることが難しいという声があった。また、Raspberry Piを利用していることも、簡単にこの教材を利用できない要因となっていた。ロボットカーのように学習者が興味を持って学習に臨め、安価に手間なくプログラミング教育用環境を準備できる教材があれば、より利用される教材になると考えられる。

そこで本研究では、安価で簡単に準備可能、かつ、学習者が興味を持ってプログラミングを学習可能な教材を開発することを目的とする。提案する教材は、ドローンを操作するプログラムを作成することでプログラミングを学習するというものである。近年、ドローンは様々な場面で利用されてきており、テレビ番組で取り上げられることもある。そのようなドローンを用

いた教材を用意し、学習者の作成したプログラムによって実際にドローンが学習者の思い通りに動作すれば、学習者は興味・関心を持ってプログラミングを学習できると思われる。また、トイドローンであれば安価に購入可能である。それに加え、リモートから送信された制御命令によって制御可能なドローンであれば、複数の学習者が1台のドローンを利用して学習する場合でも学習しやすい環境を構築できる。このように、ドローンを利用しても比較的安価にプログラミング教育用環境を構築できる。

本研究では、安価で簡単に教材を利用できるようにするために、以下のような教材を作成する。

- 必要なファイルを任意のフォルダ内に保存し、バッチファイルを実行するだけで利用開始できるような教材にする
- OS以外は無料で利用できるソフトウェアのみを利用する
- 市販の安価なトイドローンを利用し、組み立て作業なし、あるいは簡単な組み立て作業だけで本教材を利用できるようにする
- 無理なく複数の学習者が1台のドローンを利用して学習できるようにする
- ドローンを制御するためのプログラムをScratch 3.0で作成できるようにする

これらの特徴を持つ教材を開発することにより、容易に利用可能で安価な教材となる。

以降の章では、本研究で提案する教材で使用するドローン制御システムをII. で、ドローン制御プログラムの作成手順や作成方法をIII. で、本教材の利用方法をIV. で説明した後、V. で開発したドローン制御システムを使ったドローン制御実験について述べ、最後にVI. で本論文をまとめる。

II. ドローン制御システム

本研究で提案するプログラミング教育用教材のためのドローン制御システムの概要を図1に示す。本システムは、ドローン、ドローン制御用コンピュータ、ドローン制御プログラム作成用コンピュータから構成されている。以下に各構成要素について説明する。

1. ドローン

本システムでは、ドローン制御用コンピュータから送信されるドローン制御命令に従って動作させることのできるドローンを利用する。この条件を満たすドローンの1つにDJI社が製造・販売するTelloがある。

Telloには無線AP機能が搭載されており、無線LAN経由で受信したTelloの制御命令によってTelloの動作を制御可能である。そのため、本システムで使用するドローンとしてTelloを利用できる。また、Telloは13000円程度で購入でき、バッテリーを装着するだけで利用できるようになる。加えて、Telloの重量は80gであり、利用する際にドローン登録システムに登録しなくても利用可能である。このように、安価かつ容易に利用可能であるので、本教材で利用するドローンとして今回はTelloを利用する。今回使用するTelloを図2に示す。

2. ドローン制御用コンピュータ

ドローン制御用コンピュータでは、ドローン制御用拡張機能を利用できるScratch 3.0をドローン制御プログラム作成用コンピュータに提供するためのScratch 3.0用サーバー及びドローン制御プログラム作成用コンピュータから送信されるドローンへの制御命令を受信し、使用するドローンに応じてそのドローン用の制御命令の形式に変換してドローンに送信するための制

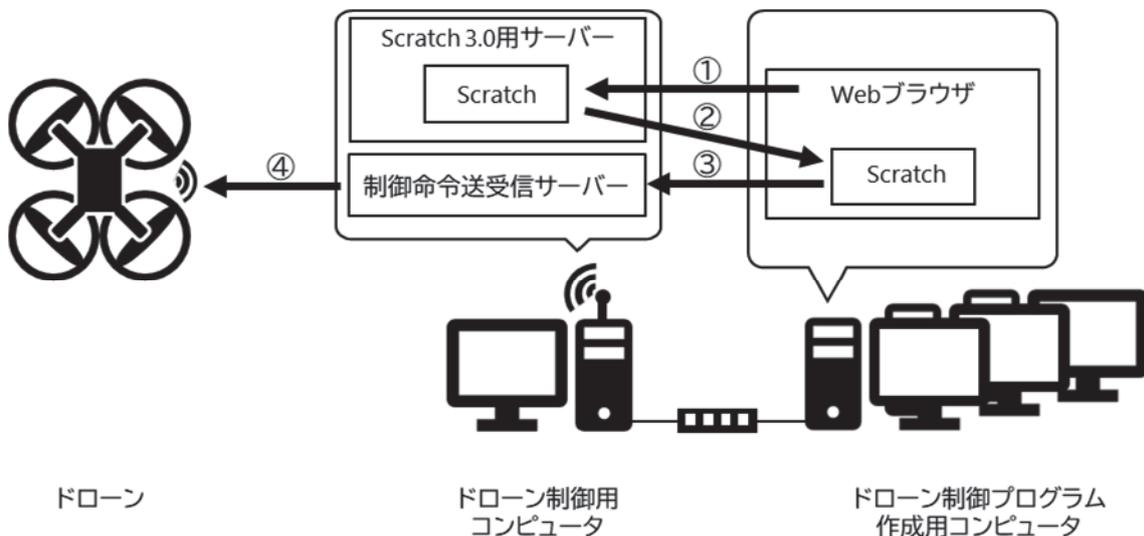


図1 ドローン制御システムの概要



図2 Tello

御命令送受信サーバーを動作させる。ここで、Scratch 3.0用サーバー及び制御命令送受信サーバーについて説明する。

(1) Scratch 3.0用サーバー

本Scratch 3.0用サーバーを用いて、ドローン制御プログラム作成用コンピュータにドローン制御用拡張機能を利用することができるScratch 3.0を提供する。

Scratch 3.0用サーバーを動作させるためには、Node.js [8] 及びScratch 3.0の開発環境 [9] [10] が必要となる。Scratch 3.0用サーバーはNode.js上で動作するため、Node.jsが動作する環境を準備する。また、ドローン制御用拡張機能をScratch 3.0で利用できるようにするために、Scratch 3.0開発環境に本研究で開発するドローン制御用拡張機能を追加し、ドローン制御用拡張機能を利用可能なScratch 3.0を提供できるようにしておく。なお、ドローン制御用拡張機能についてはIII. で説明する。このようにして準備したScratch 3.0開発環境をNode.jsで実行すると、Scratch 3.0用サーバーを起動することができる。Scratch 3.0用サーバーは、標準では8601番ポートで待ち受けるHTTPサーバーとして起動し、Scratch 3.0用サーバーにアクセスしたWebブラウザに対してScratch 3.0を提供する(図1中①②)。

(2) 制御命令送受信サーバー

制御命令送受信サーバーは、学習者が作成したドローン制御プログラムからHTTP GETメソッドにより送信される制御命令を受信し(図1中③)、その命令通りに動作するようドローンに制御命令を送信する(図1中④)機能を持っている。Telloを使用する場合、ここでTello用の制御命令をTelloに送信することになる。Tello用の制御命令は文献 [11] [12] に詳しく書かれている。

本研究では、本制御命令送受信サーバーをPythonにより作成することにした。Pythonを選択した理由は、簡単にHTTPサーバーを作成可能なライブラリがあることと、Pythonは機械学習用プログラムを作成する際によく使用されており、本教材の拡張として機械学習を導入することを想定した場合、Pythonで

あれば導入しやすいと考えたからである。

制御命令送受信サーバーがドローンへ制御命令を送信するため、ドローン制御用コンピュータはドローンと通信できる必要がある。Telloを使用する場合、Telloと無線LAN経由で通信する必要がある。有線LANデバイスと無線LANデバイスを有するノートパソコンやデスクトップパソコンで、かつ、校内ネットワークに有線LANで接続されているパソコンであれば、Telloの無線AP機能を利用できるため、ドローン制御用コンピュータとして利用可能である。無線LANデバイスを持たないデスクトップパソコンや校内ネットワークに無線LANで接続しているパソコンの場合には、USBで接続する無線LANアダプタ等を用意すれば本教材で利用できるようになる。

なお、本稿執筆時において、Telloを操作するプログラムを作成するためのScratchの拡張機能のソースコードが公開されており [11] [13]、本制御命令送受信サーバーを利用しなくてもScratchを利用してTelloの制御を行うことは可能である。しかし、それらの拡張機能はScratchから直接ドローンに制御命令を送信する仕組みになっており、1台のTelloを1人で操作することを想定して作成されている。複数の学習者が1台のドローンを利用する際には、ある学習者がドローンを利用している間に別の学習者がそのドローンを利用できないよう、排他制御をする必要がある。そうしないと、ある学習者が作成したプログラムを実行中に別の学習者が作成したプログラムからの制御命令を実行できてしまう。そのようになった場合、ドローンが動いた際に、その動作がどちらのプログラムが実行された結果なのか分からなくなってしまうため、どちらの学習者も作成したプログラムが正常に動作しているか判断できなくなってしまう。

そのような状況になることを防ぐためには、排他制御をするか、Telloを制御するプログラムを作成するためのコンピュータを複数の学習者で共有する、あるいは、誰かがTelloを使用している間は他の学習者が絶対にTelloの制御用プログラムを実行しないようにするといった工夫が学習時に必要になる。学校現場のような、多くの学習者が同時に学習する状況では、運用上の工夫なしに利用できる教材であることが望ましい。そのため、学校で利用する場合には排他制御ができるシステムであることが重要である。文献[11] [13]の仕組みの場合、ドローン側で排他制御をする必要があるが、Telloにはその仕組みがない。そのため、別の学習者が制御コマンドを送信することを防ぐことができない。Tello以外の本システムで利用できる多くのドローンも排他制御はできないと思われる。

本ドローン制御システムであれば、複数のドローン制御プログラム作成用コンピュータから送信されるドローン制御命令を制御命令送受信サーバーで受信する

ようにしているため、制御命令送受信サーバーで排他制御を行い、ある学習者のドローン制御プログラムの実行中に他の学習者の制御命令をドローンに送信しないようにできる。そのようにすることで、複数の学習者が1台のドローンを利用する場合においても、学習者全員がプログラミングを学習しやすい環境を運用上の工夫なしに実現できる。

3. ドローン制御プログラム作成用コンピュータ

学習者は、ドローン制御プログラム作成用コンピュータでScratch 3.0を利用してプログラムを作成する。このコンピュータは、Scratch 3.0の動作するWebブラウザが動作すればよく、多くのパソコン、タブレット、スマートフォンをドローン制御プログラム作成用コンピュータとして利用できる。

複数の学習者が同時に1台のドローンを利用してプログラミングの学習をする際には、ドローン制御プログラム作成用コンピュータを各学習者の手に用意し、各学習者の使用するコンピュータでプログラムを作成して実行する。なお、学習者が1人しかいない場合には、ドローン制御用コンピュータとドローン制御プログラム作成用コンピュータを1台のコンピュータで兼ねることも可能である。

ドローン制御プログラム作成用コンピュータを用いたドローン制御プログラムの作成環境の準備手順やプログラムの作成方法については次章で説明する。

III. ドローン制御プログラムの作成

本教材を利用してプログラミングを学習する場合、学習者はまずドローン制御プログラム作成用コンピュータでWebブラウザを起動し、ドローン制御用コンピュータ上で動作するScratch 3.0用サーバーにアクセスし（図1中①）、Scratch 3.0をWebブラウザ上に表示する（図1中②）。次に、Scratch 3.0の画面左下に表示される「拡張機能の追加」ボタンをクリックしてドローン制御用拡張機能をScratch 3.0に追加する。図3に「拡張機能の追加」ボタンを示す。図中左下に表示されている青いボタンがそれである。「拡張機能の追加」ボタンをクリックすると、拡張機能の一覧が表示される。一覧中に表示されるドローン操作拡張機能を図4に示す。一覧中のドローン操作拡張機能をクリックすると、ドローン操作拡張機能がScratch 3.0に追加され、ドローン制御用ブロックをScratch 3.0で利用できるようになる。Scratch 3.0に追加されるドローン制御用ブロックを図5に示す。追加されたブロックを使用すればドローン制御プログラムを作成することができる。

ドローンの動作を制御するプログラムを作成する際には、最初に離陸ブロックを実行する。離陸ブロック



図3 「拡張機能の追加」ボタン（図中左下）

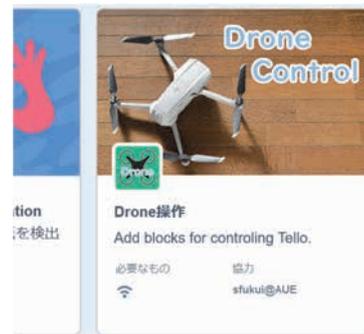


図4 ドローン操作拡張機能



図5 ドローン制御用ブロック

が実行されると、ドローンが離陸して空中に静止する。離陸ブロックが実行されたタイミングで排他制御されるようになり、着陸ブロックが実行されるまで、あるいはドローン制御用ブロックのいずれかが実行されてから2分間、他の学習者がドローン进行操作できないようになる。離陸ブロックを実行後、移動用ブロックと回転用ブロックを実行できるようになる。移動用ブロックは、上下左右前後のいずれかの方向にドローンを移動させることが可能なブロックで、どの方向にどれくらい移動させるか設定できる。回転用ブロックは、ドローンをその場で右回転か左回転させることができるブロックであり、どの方向に何度回転させるか設定できる。これらのブロックを利用してドローンの動作を制御する。プログラムの最後には着陸ブロックを実行してプログラムを終了する。このブロックが実行されると、ドローンが着陸し、ドローンの制御を終了す

る。着陸ブロックが実行されると排他制御が解除され、他の学習者がドローンを利用できるようになる。

ドローン制御用ブロックがScratch 3.0で実行された場合、実行されたブロックに応じたURLで制御命令送受信サーバーにHTTP GETメソッドでアクセスする。離陸ブロックが実行された場合を例として説明すると、離陸ブロックが実行された際には、http://(ドローン制御用コンピュータのIPアドレスかホスト名):8080/land にアクセスする。なお、この例は、制御命令送受信サーバーが待ち受けるポート番号を8080番とした場合の例である。

IV. 本教材の利用方法

本章では、本教材の利用方法を説明する。本教材を利用するためには、事前に本教材に必要なファイルをドローン制御用コンピュータでダウンロードした後、Scratch 3.0用サーバーと制御命令送受信サーバーを起動できるように準備をする必要がある。簡単に準備を終えられるよう、必要なファイルをダウンロードした後、バッチファイルを実行するだけでほぼすべての準備が終わるようにしてある。両サーバーの起動準備を1回しておけば、以降はそれらのサーバーを起動するためのバッチファイルを実行するだけで本教材を利用できる。

ここで、ドローン制御用コンピュータで行う準備について詳しく説明する。まず、本教材を利用するための準備に必要なバッチファイルや本教材利用時に必要なファイル一式がアーカイブされたファイル（以下、本教材用ZIPファイルという）をダウンロードし、任意のフォルダ内に展開する。次に、Node.jsの実行に必要なファイル一式がアーカイブされたZIPファイル、Scratch 3.0の開発に必要な2つのZIPファイル、Pythonの実行に必要なファイルを公式Webサイトからダウンロードし、本教材用ZIPファイルを展開してできたフォルダ内にそれらのファイルを保存する。その後、同フォルダ内に存在する環境準備用のバッチファイルを実行する。その後は、後述する「Extract」ボタンを1回クリックした後、処理が終わるのを待っているだけで準備を終えることができる。

本教材を学校で使用するためには、ソフトウェアを自由にインストールできない環境でも利用できる必要がある。そのため、Scratch 3.0用サーバーと制御命令送受信サーバーを動作させるために必要なソフトウェアをインストールすることなく利用できるものにならなければいけない。Node.jsはインストールしなくても動作させることが可能である。Node.jsを利用するために、インストール用ファイルではなくZIPファイルを公式サイト [8] からダウンロードする。また、Scratch 3.0の開発環境の構築のためのファイルも公

式サイト [9] [10] からZIPファイルをダウンロードして利用する。Scratch 3.0の開発環境の構築にはgitが用いられることが多いが、gitを利用できない学校が多いと考えられるため、ZIPファイルをダウンロードして使用する。Pythonにもインストールせずに動作させることのできるパッケージが複数ある。本教材では、WinPython [14] を利用する。WinPythonを動作させるために必要なファイル一式がパッケージングされた自己解凍ファイルを公式サイト [14] からダウンロードし、それを利用する。なお、自己解凍ファイルを利用するため、環境準備用バッチファイルを実行すると、実行後すぐにWinPython用自己解凍ファイルを展開するためのダイアログボックスが表示される。その際、作業を続けるために「Extract」ボタンをクリックする必要がある。

本章で挙げた、本教材で利用するすべてのソフトウェアやファイルは無料で利用できる。本教材を利用するために学校に設置済みのコンピュータを利用するのであれば、ドローンを購入するだけで本教材を利用することができる。そのため、本教材を安価に利用することが可能である。

V. 実験

開発したプログラミング教育用環境が正常に動作するか確認するための実験を行った。実験には、ドローン制御用コンピュータ兼ドローン制御プログラム作成用コンピュータとしてWindows 10が動作するノートパソコンを利用した。

まず、本研究で作成した複数のバッチファイルが正常に動作し、簡単に本教材を利用することができるか確認した。最初に、本教材用ZIPファイルをデスクトップ上に展開した。次に、本教材に必要なファイルを公式サイトからダウンロードし、展開したフォルダ内にそれらのファイルを保存した。その後、本教材の環境準備用バッチファイルを実行した。かなり時間を要したが、バッチファイル実行後すぐに表示される「Extract」ボタンを1回クリックするだけで準備が終わった。これにより、簡単に本教材を使用するための準備が終わることが確認できた。なお、今回の実験のためにダウンロードしたNode.jsとWinPython用のファイルは、node-v16.17.0-win-x64.zip及びWinpython64-3.10.5.0dot.exeである。それらは本稿執筆時におけるそれぞれのソフトウェアの安定版の最新バージョンのファイルである。準備終了後、バッチファイルを実行することによってScratch 3.0用サーバーと制御命令送受信サーバーを動作させることができるか確認した。バッチファイルを実行すると、2つのサーバーが動作した。2サーバーの初回起動時には「Windowsセキュリティの重要な警告」が表示されるが、「アクセ

スを許可する」ボタンをクリックすることにより、正常に起動することを確認した。これらの結果から、本教材を簡単に利用できることが確認できた。

次に、ドローン制御用コンピュータと同じノートパソコン上でWebブラウザを起動し、Scratch 3.0用サーバーにアクセスしてScratch 3.0を表示した。そのScratch 3.0によってドローン制御用プログラムを作成し、Telloがプログラミングされた通りに動作するか確認した。上へ20cm移動させるブロックを実行した際には、Telloが上へ約20cm移動した。90度右回転するブロックを実行した際には約90度右回転した。複数のブロックを組み合わせたプログラムを実行した際にも概ね想定通りの動作をすることを確認した。これらの結果から、ドローン操作用拡張機能を用いてTelloを制御可能であることを確認した。ただし、本実験を行った際、周りに家具や壁等が存在するとTelloが空中で安定しないことがあった。これは、Telloが小さく軽いため、Tello自身の起こした風によって動作が不安定になってしまったものと考えられる。本ドローン制御システムとTelloを使用してプログラミング教育を行う場合には、体育館などの広い空間で行うのがよいことが分かった。

また、本教材の追加機能として、ドローンに搭載されているカメラ映像をScratch 3.0の画面上に表示できるようにした。更に、機械学習に関する教材としても利用できるよう、QRコードや音声によってドローンを操作できるように改良した。これらの機能追加によりScratch 3.0に追加されたブロックを図6に示す。

ドローンから得られる映像を表示することにより、ドローン操作の臨場感が増したように思われる。しかし、この機能を実装した結果、制御命令送受信サーバーが安定して動作しなくなってしまう。Telloのカメラからの映像を取得できないことが稀にあり、そうした際に制御命令送受信サーバーがダウンすることがあった。画像表示機能を提供するためには、この問題を解決する必要がある。

音声による操作では、「上」、「下」、「右」、「左」、「前」、



図6 実験で新たに追加したブロック

「後ろ」という命令を用意し、それらの命令の発声によってその命令通りに動作するようにした。各命令が実行された際のドローンの移動距離を20cmとした。実際に動作させることにより、音声による操作も可能であることを確認した。今回の実験では、音声認識にWeb Speech APIを利用したが、日本語で操作する場合には認識精度がWebブラウザによってかなり異なることが分かった。Microsoft Edge, Google Chrome, Mozilla Firefoxで実験した結果、Google Chromeが今のところ日本語で操作するには最も適していることが分かった。

QRコードによる操作実験も行った。制御命令がエンコードされたQRコードをドローンに搭載されているカメラで撮影し、制御命令の通りにドローンが動作するか確認した。この実験のためにQRコードとそれに埋め込まれている命令が何か人間が分かるようにするための命令（以下、命令語という）の2つが印刷されたプリントを用意した。用意したプリント例を図7に示す。用意したプリントをドローンで撮影することで、命令語通りにドローンが動作し、QRコードによってドローンを制御できることを確認した。これにより、ドローン制御プログラム作成用コンピュータを用意しなくてもドローンを制御できるようになり、アンプラグドプログラミングによるプログラミング学習を行うこともできるようになった。

ここで、QRコードによるドローン操作を可能にすることによって実現できる、プログラミング教育用途ではない教材としての使用方法も提案する。この教材では、QRコードに埋め込まれている命令と命令語をわざと一致させずに作成したプリントも用意する。最初は命令語通りに動作するプリントを利用して本教材利用者にドローンを動作させ、命令通りにドローンが動作することを確認させる。次にQRコードの命令と命令語が一致しないプリントを使ってドローンを操作させる。この場合、最初のうちはドローンが命令語と異なる動作をするため利用者は戸惑うであろうが、徐々に慣れていき、プリントの規則性を理解して利用



上

図7 QRコードによる制御用プリント例（QRコードにはUPというドローン制御命令が埋め込まれている）

者の思った通りの動作をさせることができるようになると思われる。そのような体験をした後、なぜ前者のプリントでは命令通り動作し、後者のプリントでは命令通り動作しないのか利用者に考えさせる。この体験を通して、人間は場面に応じて臨機応変に対応可能であるが、コンピュータはプログラムされた通りにしか動作しないということを利用者に体験させることが可能である。このように、コンピュータの特性の1つを理解させることができる教材として利用することも可能であると考えられる。

VI. おわりに

本研究では、安価かつ誰でも簡単に準備して利用することができるドローンを用いたプログラミング教育用教材を開発した。

本教材は、システムを動作させるために必要なファイルを任意のフォルダに保存し、バッチファイルを実行するだけで使用準備をほぼ終わることができる。かつ、バッチファイルを実行するだけで教材の利用に必要なプログラムを実行できる。これらの結果、利用しやすい教材となった。また、使用可能なドローンの条件はあるものの、安価なドローンを使用できる教材にした。かつ、排他制御のできるシステムを開発し、複数の学習者が1台の安価なドローンを利用してプログラミングを学習しやすくなるようにした。そのため、学校等で多数の学習者がプログラミング学習をする際、比較的安価に学習用環境を整備することが可能となった。加えて、プログラミング初学者でも容易に利用可能なScratch 3.0によってドローン制御用プログラムを作成・実行できるようにした。これにより、学習者が学習時に困難さをあまり感じることなくプログラミングを学習できるようになった。これらの成果により、ドローンを実際に制御する体験を通して学習者がプログラミングを楽しく学習できる教材を手間なく安価に利用できるようになった。

今後の課題として、本教材を利用したカリキュラム

の作成およびドローンに搭載されているカメラの映像をScratch 3.0の画面上に表示した際の制御命令送受信サーバーの動作の安定性の向上が挙げられる。

文献

- [1] 文部科学省, “平成 29・30・31年改訂学習指導要領”, https://www.mext.go.jp/a_menu/shotou/new-cs/1384661.htm (2022年8月30日 閲覧)
- [2] コエテコ by GMO, “コエテコx船井総研「2018年子ども向けプログラミング教育市場調査」を実施-2023年の市場規模は226億円に拡大すると予測”, <https://coeteco.jp/articles/10180> (2022年8月30日 閲覧)
- [3] 文部科学省, “教材情報”, 小学校を中心としたプログラミング教育ポータル, <https://miraino-manabi.mext.go.jp/teaching> (2022年8月30日 閲覧)
- [4] 日経クロステック, “ラズパイで作ろう! ゼロから学ぶロボット製作教室”, <https://itpro.nikkeibp.co.jp/atcl/column/15/040800081/> (2022年8月30日 閲覧)
- [5] 福井真二, 藤澤悠貴, “Raspberry Piを用いたプログラミング教育用教材の開発”, 愛知教育大学研究報告自然科学編, 第69輯, pp. 23-27, 2020
- [6] 福井真二, “Scratch 3.0を用いたプログラミング教育用ロボットカーの制御システムの開発”, 愛知教育大学研究報告自然科学編, 第71輯, pp. 24-29, 2022
- [7] Scratch財団, “Scratch”, <https://scratch.mit.edu/> (2022年8月30日 閲覧)
- [8] Node.js, <https://nodejs.org/ja/> (2022年8月30日 閲覧)
- [9] scratch-vm, <https://github.com/llk/scratch-vm> (2022年8月30日 閲覧)
- [10] scratch-gui, <https://github.com/llk/scratch-gui> (2022年8月30日 閲覧)
- [11] 高橋隆雄, “Scratchでトイドローンをプログラミングして飛ばそう”, 秀和システム, 2018
- [12] Ryze Tech., “Tello SDK 2.0 User Guide”, <https://dl-cdn.ryzerobotics.com/downloads/Tello/Tello SDK 2.0 User Guide.pdf> (2022年8月30日 閲覧)
- [13] Kanta Ebihara, “scratch3-tello”, <https://github.com/kebhr/scratch3-tello> (2022年8月30日 閲覧)
- [14] WinPython, <https://sourceforge.net/projects/winpython/> (2022年8月30日 閲覧)

(2022年9月26日受理)