

# 幼児教育の現場における人型ロボットの活用Ⅱ

—手遊びを題材とした Pepper 教育アプリケーションの開発—

奥 恵・渡 邊 裕

## Utilization of Humanoid Robots in Preschool Education, Part 2

— Development of an Educational Application Based on Hand Play for Pepper —

OKU Megumi, WATANABE Hiroshi

キーワード：手遊び、幼児教育、Pepper、人型ロボット、アプリケーション開発

### 1. はじめに

近年、人型をした家庭用ロボットの開発が急速に進み、近い将来ロボットが家庭や教育の現場に普及した場合にどのような教育的効果をもたらすのかについては興味のある課題である。ロボット技術は今後ますます進展することが期待されるが、それと並行して人型ロボットの教育コンテンツ開発や教育の現場における人とロボットとの相互作用の研究についても今後詳細に検討していく必要がある。

Pepper<sup>1)</sup>で動作する子どもを対象とした教育アプリケーションの例としては、ロボットの身体性を活かした色やからだの英語名を学ぶアプリコンテンツの開発例 (Tanaka *et al.* 2015) や、

Pepper に体操をさせるアプリケーション開発の例 (阿久津・野田 2016)、手作り紙芝居を題材としたコンテンツの試作例 (渡邊 2020) などがある。

本研究では人型ロボット Pepper で動作する幼児教育向けコンテンツの例として手遊びの実演に着目し、実際に幼児保育学科の教員が演じる手遊びの動作を分析した上で、それらの動作を実現するためのコンテンツ開発の実践例について報告する。

### 2. 研究の方法

#### 2.1. アプリケーション開発環境

埼玉東萌短期大学では Pepper を学校現場においてプログラミング教育等での活用を想定した Pepper 社会貢献プログラム<sup>2)</sup>に参加し、教育研究開発を目的に 2019 年 4 月～2022 年 3 月までの 3 年間、Pepper の貸与を受けている。今回は Pepper を動作させるためのアプリケーションであるロボアプリ<sup>3)</sup>を開発するための統合開発環境として、Choregraphe 2.5.5<sup>4)</sup>を利用した。その操作画面を図 1 に示す。Choregraphe の基本的な操作方法については村山ほか (2015) 等の解説書がある。

Pepper の基本動作は画面左下のボックスライブラリから選ぶことができる。ボックスライブラリには「Move To」(移動する)、「Set Language」(言語設定を変更する)、「Say」(発話

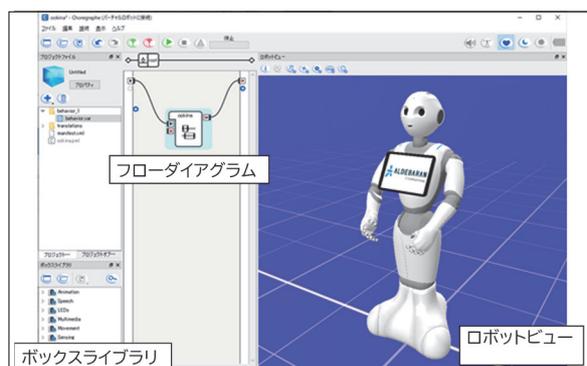


図 1 Choregraphe の操作画面

する)、「Animated Say」(動きをつけながら発話する)等、さまざまな基本動作が登録されている。これらのボックスを画面中央のフローダイアグラムに配置して、それらを左上の「on Start」入力から線をつないでいき、フローを作成する。最後に右上の「on Stopped」出力に接続すればフローダイアグラムが完成する。

このように Choregraphe には標準的なボックスライブラリが搭載されていることから、それらを組み合わせることで各種の動作を行うことが可能である。例えば、「Go to Position」ボックスに数種類のポーズを登録し、それらフローダイアグラムに配置して並べ、フローをつなげていけば、ポーズを順に変えていく動作を実現できる。ポーズとポーズの間は自動的に補完されてスムーズに動くようになる。しかしこの方法では、ポーズとポーズのタイミングを制御することができない。そこで今回は、タイミングの制御が可能なタイムラインボックスを用いてモーションを作成することにした。

## 2.2. ポーズの作成

タイムラインボックスでモーションを作成するためには、まず動作のキーとなるポーズを作成し登録する必要がある。ポーズとは静止状態での Pepper の姿勢のことである。

ポーズを作る方法は2つあり、Choregraphe 上の3Dバーチャルロボットで作成する方法と、Choregraphe 上で実物の Pepper に接続し、アニメーションモードで Pepper の実物を手で動かしながらポーズを決め、それを Choregraphe に登録する方法である。簡易な方法は后者であるが、Pepper との接続環境を整える必要がある。そこで今回は、Choregraphe 上のバーチャルロボットでポーズを作成した。

ポーズの作成画面を図2に示す。上のロボットビュー画面で、変更したい部位を選択すると、下のインスペクタ画面で関節の角度を変更することができる。また、ロボットビュー画面に表示される円形のローテーションハンドルをマウスで回転

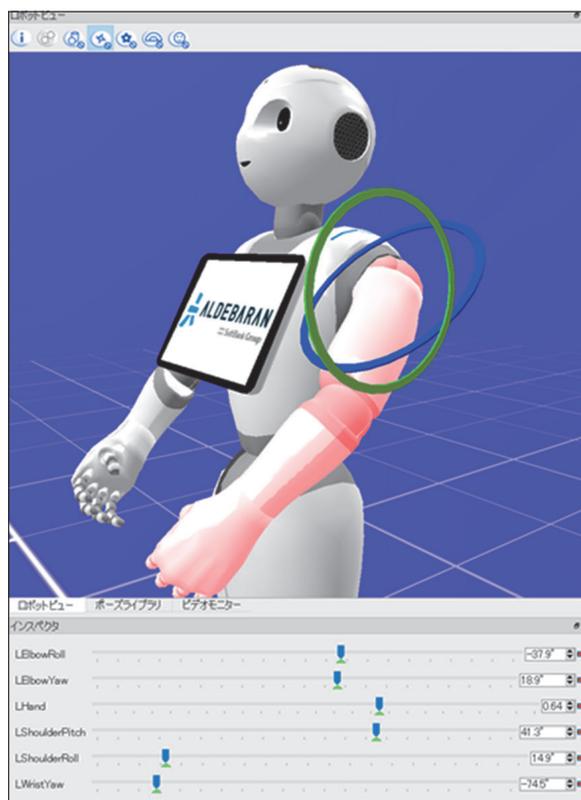


図2 バーチャルロボットによるポーズ作成

させることにより、関節の角度を調整することもできる。Pepper には人の関節に相当するアクチュエータが配置されており、各アクチュエータの値をインスペクタ画面で設定することにより関節の角度を決めていく。

ただし、変更できる関節の場所や角度の範囲はあらかじめ決められていることや、自分の身体に自身を衝突させないようにするための衝突防止機能が備わっているため、おのずと作成できるポーズや動作には制限がかかる点に注意しなければならない。

作成したポーズはポーズライブラリに登録する。一つのモーションを作るためには、キーとなるポーズを複数個作成し、登録しておく必要がある。

## 2.3. モーションの作成

ポーズの登録をしたら、フローダイアグラム上で新規にタイムラインボックスを作成する。ボックスにはタイムラインの名前を付けることができる。タイムラインにはフレーム番号が付けられており、デフォルトでは1秒が25フレームに設定

表1 「おおきなくりのきのしたで」のキーフレーム

場面	発話	開始時刻 秒	フレーム 番号
(開始)		0.00	0
A01	おおきなくりのきのしたで	1.41	35
A02	おお	4.76	119
A03	きな	5.93	148
A04	くりの一	6.9	173
A05	きの	8.17	204
A06	した	8.9	223
A07	でー	9.69	242
A08	あーなーたーと	10.79	270
A09	わーたーしー	13.52	338
A10	なーかー	16.27	407
A11	よーくー	17.86	447
A12	あそびま	19.27	482
A13	しょー	20.34	509
A14	おお	21.69	542
A15	きな	22.56	564
A16	くりの一	23.55	589
A17	きの	24.38	610
A18	した	25.07	627
A19	でー	26.07	652
(終了)		28.03	701

されており、今回はこの値を用いている。

図3にタイムラインの設定例を示す。横軸は時間軸を表す。例えばA1～A8までの8つのポーズをタイムライン上に登録していくことができる。最後に終了フレームを設定すれば完了である。このようにして作成したタイムラインボックスは、フローダイアグラム上で別の動作と並列に実行することができる。例えば、発話をさせながらモーションを実行することも可能である。

### 3. 手遊びモーションの開発

#### 3.1. 開発する手遊びとキーフレームの選定

今回のアプリケーション開発では、以下の3つ

の手遊びを選定した。いずれも幼児教育の現場ではよく演じられる手遊びである。

- (1) おおきなくりのきのしたで
- (2) むすんでひらいて
- (3) グーチョキパーでなにつくろう

手遊びの動きを実現するためには、Pepper自身が衝突したり早い動きなどは実現させることが難しい。手遊びでよく見受けられる手をたたく動作は両手の手のひらの衝突が伴うため、コンテンツ開発の演目からはなるべく避けることにした。

Pepperのモーションを開発するためには、キーフレームとなるポーズを決める必要がある。そこで著者の一人が実際に手遊びを実演し、その様子を動画撮影することにより手遊びの動きのデータを取得した。これらの動画から動きの中でキーフレームとなる場面を抽出した。この結果、「おおきなくりのきのしたで」は19場面、「むすんでひらいて」は12場面、「グーチョキパーでなにつくろう」は16場面のキーフレーム画像を選定した。

#### 3.2. 「おおきなくりのきのしたで」の手遊びモーションとその特徴

表1に「おおきなくりのきのしたで」の手遊びでキーフレームとして選定した場面とその場面における発話、場面の開始時刻 [秒] とタイムライン上でのフレーム番号を示す。場面数は全19場面であり、時間は約28秒である。これらのキーフレーム画像と、この画像を元にChoregraphe上で作成したPepperのポーズを図4に示す。

タイムラインのプログラム上では、A01の開始場面は1.41 [秒] (35フレーム) 経過した後から始めている。この理由は、Pepperが各関節をモーターで駆動させて動きを付けているため、あまりにも急激な変化を起こさせないように、最初の

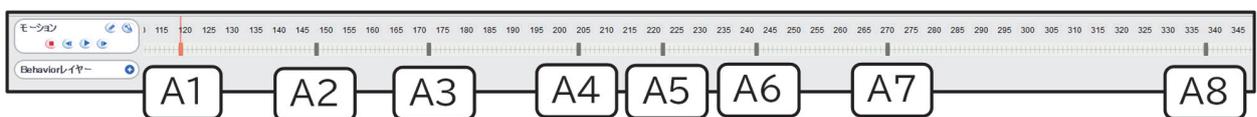


図3 タイムラインの設定例

場面	人	Pepper	場面	人	Pepper
A01			A07		
(はじめ) おおきなくりのきのしたで			でー		
A02			A08		
おお			あーなーたーと		
A03			A09		
きな			わーたーしー		
A04			A10		
くりのー			なーかー		
A05			A11		
きの			よーくー		
A06			A12		
した			あそびま		

図4 「おおきなくりのきのしたで」の動作と Pepper のポーズ

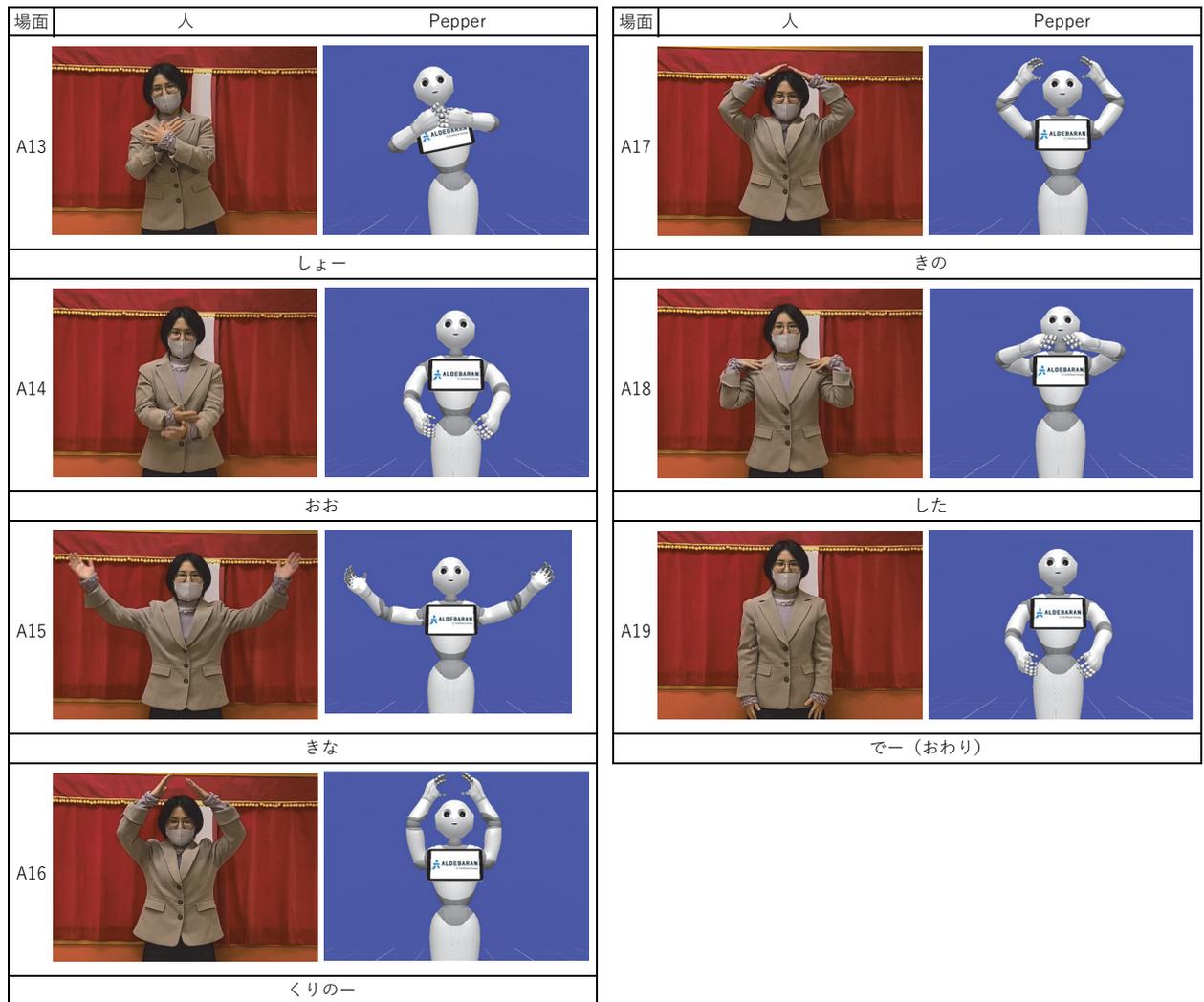


図4（つづき）「おおきなくりのきのしたで」の動作と Pepper のポーズ

キーフレームを約1秒経過した後に設定させているためである。A01では手遊びのタイトルを発話するための姿勢を取っており、実際の手遊びの動作はA02から始まる。

A02～A04までは両手を挙げる動作であるが、その後A05の場面で両手を頭に付ける動作が入る。しかし、Pepperは自身への衝突を防ぐ機能が備わっているため、実際に両手を頭につけることはできない。また、この状況はA06で両手を両肩に触れる場面でも同様である。さらにA06で人の肘の角度が鋭角となるが、Pepperはこの部分の関節角度が最大でも90°までという制限があり、完全にポーズを再現することはできない。

その後、A08の「あーなーたーと」と手を差し伸べる場面は容易であるが、A09で「わー

たーしー」と人差し指で顔を指すことはできない。Pepperの指の動作は、握るか開くかの動作はできるものの、それぞれの指を単独に動かすことができないためである。またA11で両手を胸の前で交差させる動作についても、衝突防止機能のために実現できない。

一方で、A12とA13で身体の上体を左右に動かす動作は可能である。ただし、腕の広がりによって重心の位置が不安定になる場合には、身体を倒すことができない場合がある。

A14～A19の動作は、A02～A07と基本的に同じ動きであり、モーションとしても繰り返しを適用することで容易に作成することができる。

表2 「むすんでひらいて」のキーフレーム

場面	発話	開始時刻 秒	フレーム 番号
(開始)		0.00	0
B01	むすんでひらいて	1.28	32
B02	むーすーんーで	5.48	137
B03	ひーらーいーて	8.24	206
B04	てーを	10.86	272
B05	うって		
B06	むーすんで	13.65	341
B07	まーたひらいて	16.31	408
B08	てーを	19.34	484
B09	うって		
B10	そーのーてーを	21.76	544
B11	うーえーに	25.03	626
B12		29.82	746
(終了)		32.01	800

### 3.3. 「むすんでひらいて」の手遊びモーシオンとその特徴

表2に「むすんでひらいて」の手遊びでキーフレームとして選定した場面とその場面における発話、場面の開始時刻 [秒] とタイムライン上でのフレーム番号を示す。場面数は全12場面であり、時間は約32秒である。これらのキーフレーム画像と、この画像を元にChoregraphe上で作成したPepperのポーズを図5に示す。

B01の開始場面は1.28 [秒] (32フレーム) 経過した後であり、実際の手遊びの動作はB02から始まる。B02とB03の両手をグーとパーにする動作は容易であるが、B04とB05の「てーを」「うって」の場面で実際に手をたたく動作をすることはできない。両手を近づけることは可能であるが、Pepperには胸の正面にディスプレイが付いており、人のように両肘を近づける動作をすることが困難である。そのためにB05のように両手の指の先をつけるようなポーズになってしまう。同様の現象が繰り返しの場面であるB09とB10でも発生する。しかしB10やB11のように、両手を上に上げる動作は比較的簡単である。

なお、今回の人による実演では、B02の場面で両手を握った状態で、リズムに合わせて手の拳を前後に4回揺らす動作が入っている。同様にB03

表3 「グーチョコキパーでなにつくろう」のキーフレーム

場面	発話	開始時刻 秒	フレーム 番号
(開始)		0.00	0
C01	グーチョコキパーでなにつくろう	1.24	31
C02	グー	6.72	168
C03	チョコキ	7.38	185
C04	パーで	8.00	200
C05	グー	9.31	233
C06	チョコキ	10.01	250
C07	パーで	10.72	268
C08	なにつく	12.31	308
C09	ろー	13.38	335
C10	なにつく	14.65	366
C11	ろー	15.83	396
C12	みぎてはパーで ひだりても	18.17	454
C13	パーで	21.03	526
C14	ちょーちょ	22.17	554
C15	ちょーちょ	24.58	615
C16		29.03	726
(終了)		31.89	797

でも開いた手を小刻みに前後に4回揺らしている。さらにB04とB05で両手をたたく場面では、リズムに合わせて4回の手をたたく動作をしている。今回のモーシオンでは、このような小刻みな動きは実装させていない。その理由は、Pepperの関節はモーターで稼働しているため、そのレスポンスにタイムラグが生じることがあり、ある程度早い動きになると実現できないためである。

### 3.4. 「グーチョコキパーでなにつくろう」の手遊びモーシオンとその特徴

表3に「グーチョコキパーでなにつくろう」の手遊びでキーフレームとして選定した場面とその場面における発話、場面の開始時刻 [秒] とタイムライン上でのフレーム番号を示す。場面数は全16場面であり、時間は約32秒である。これらのキーフレーム画像と、この画像を元にChoregraphe上で作成したPepperのポーズを図6に示す。

C01の開始場面は1.24 [秒] (31フレーム) 経過した後であり、実際の手遊びの動作はC02から始まる。この手遊びをPepperに動作させるためには幾らかの問題点がある。まず、C03やC06

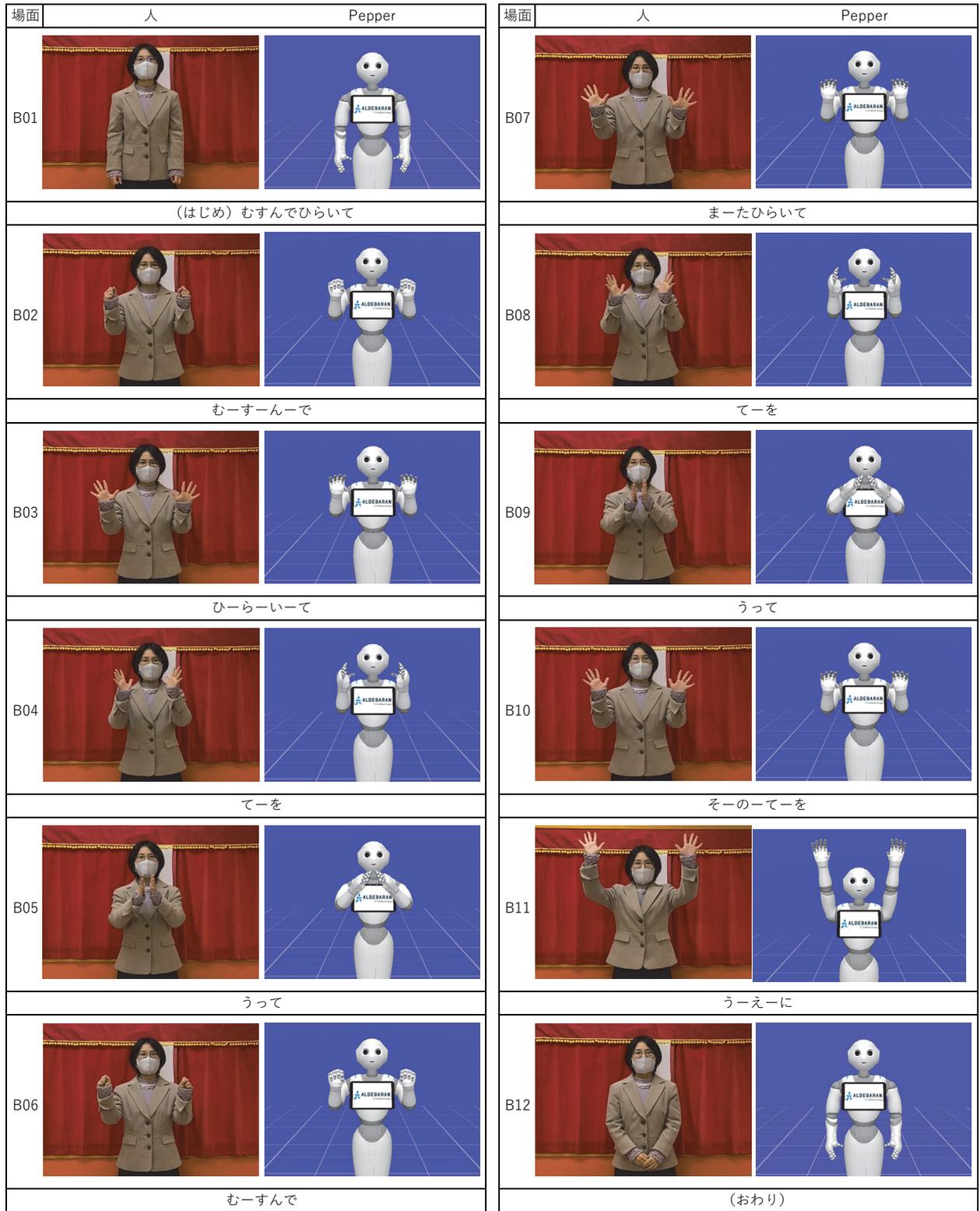


図5 「むすんでひらいて」の動作と Pepper のポーズ

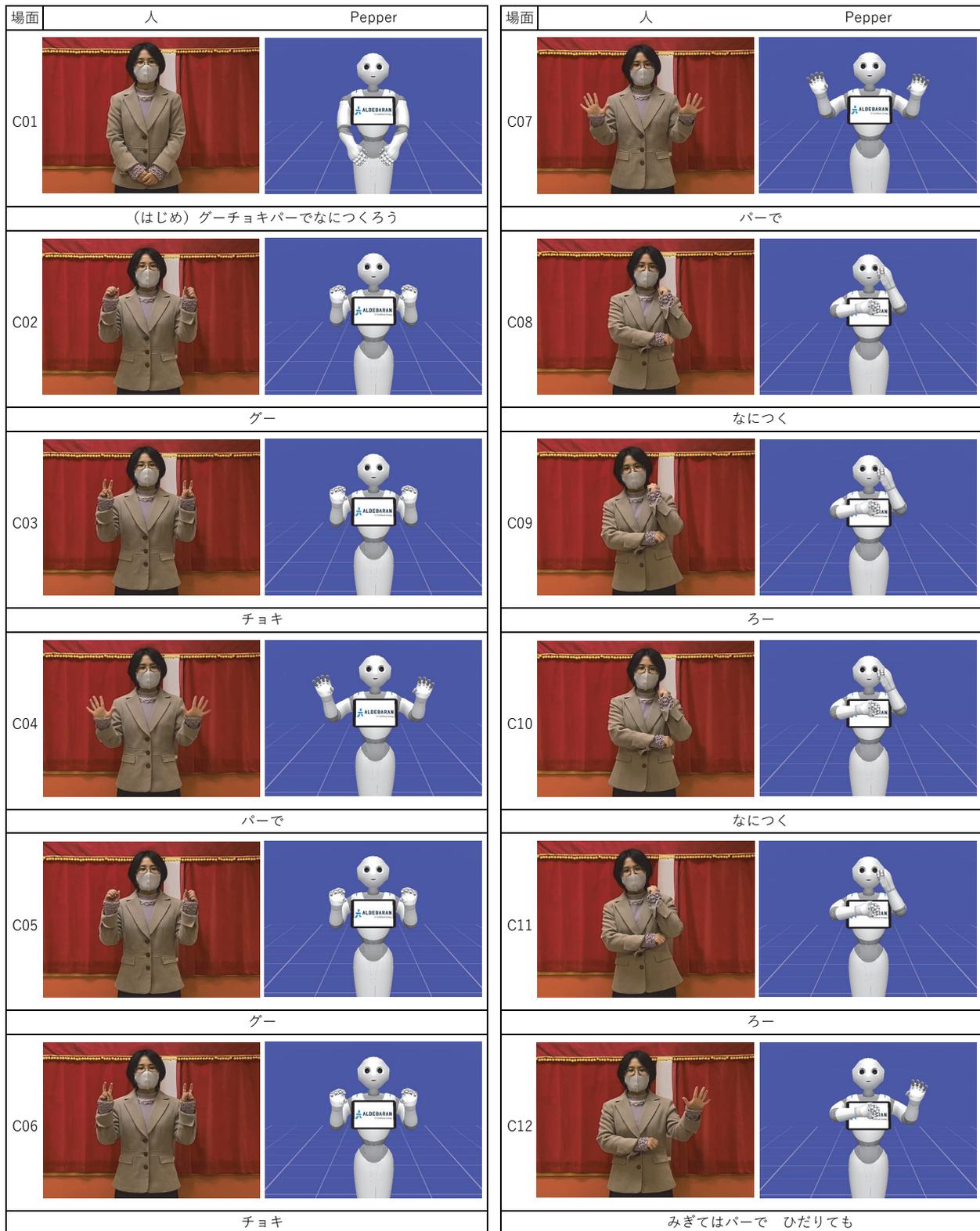


図6 「ぐーちょきぱーでなにつくろう」の動作と Pepper のポーズ

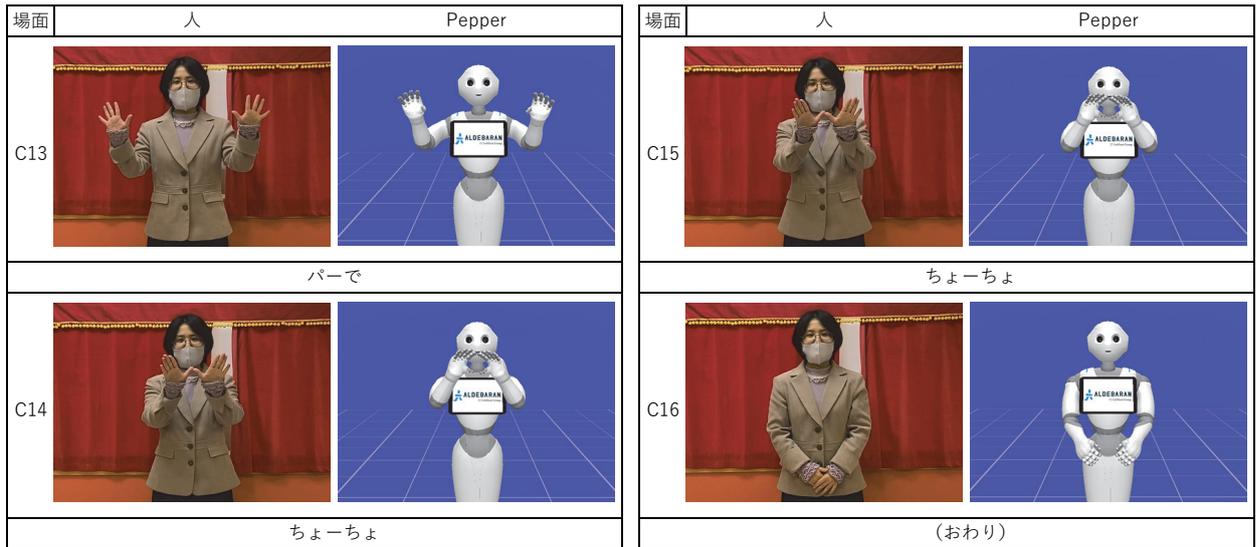


図6 (つづき) 「グーチョコキパーでなにつくろう」の動作 Pepper のポーズ

場面における「チョコキ」を Pepper の構造上、動作させることができない。次に C08～C11 にかけて状態を左右に揺らす場面があるが、Pepper の腕が左右で非対称であるため、重心のバランスがとりにくくなり、腰を左右に動かすことが出来ない。そして、C14 と C15 で両手でちょうちょの形を作る場面があるが、手の関節の構造上、今回の腕の角度では親指同士を付けることができず、結果として蝶々の形には見えなくなってしまっている点が挙げられる。

#### 4. 考察

Pepper は人型ロボットとして開発され、人間と同じように関節の部分にアクチュエータ（モーター）が配置されている。このため、両手（肩、肘、腕、手）や首、腰を動かすことが可能であり、手遊びの動作の中でも比較的簡単に再現させることができる場合がある。

しかし、今回の手遊びモーションの開発により、人が実演すると簡単なように思える動作であっても、実際に Pepper に実演させると表現することができなかつたり、動作させることが不可能なものがあることが明らかになった。例えば、以下のような制限が認められる。

- ・自身への衝突を避けるため、手をたたくなどの

接触を伴う動作をすることができない。

- ・各関節には可動範囲の角度があり、たとえば鋭角に肘を曲げるなどの動作はできない。
- ・胸の前にディスプレイが付いており、この部分に腕が衝突しないようにするため、腕の動きに制限がある。
- ・両手の動作は5本の指を同時に開くか握るかであり、それぞれの指を独立して動かすことはできない。
- ・上体の姿勢を保ち、Pepper が倒れないようにするために、身体全体を左右に振る動作には制限がある。
- ・モータに負荷がかかるような急激な動きや早い動きは実現できない。

一方で、Pepper に手遊びを実演させる場合に、以下のようなロボットならではのメリットも認められる。

- ・手遊びのような繰り返しの多い動作を実現させることが容易である。一つの手遊びであっても、キーとなるポーズを決めてしまえばそれを繰り返してプログラミングすることが可能である。また、動作と発話は分離できるため、発話の内容を変えて1番、2番、3番というように同様のモーションで手遊びを実現させることも可能である。
- ・手遊びのアプリケーション開発では手をたたく

などの特有の動作があるが、それらのモーショ  
ンを Choregraphe のタイムラインボックスで  
一度作ってしまえば汎用的に利用できるという  
メリットがある。

## 5. 今後の課題

本研究では、幼児教育の現場でよく知られて  
いる手遊びの様子を演じた動画を撮影し、その  
動作を人型ロボット Pepper に実演させるための  
アプリケーションを開発した。今回の試みによ  
り Pepper の動作には一定の制限があることが明  
らかになったが、逆にこれらのこの制限を認識し  
た上で、今後 Pepper が実演可能なオリジナルの  
手遊びを考案していくことは可能であると思われ  
る。また、今回はモーシヨンの再現にとどまっ  
ているが、発話や音を同時に流すことも可能である。  
Pepper が人に対して効果的な手遊びを実演する  
ことができるようになれば、その実演をみた学生  
が手遊びを Pepper から学ぶといった場面を設定  
し、その教育的効果を検証するという研究の方向  
性についても検討していきたい。

## 引用文献

- 阿久津由嗣, 野田夏子 (2016) Pepper に体操さ  
せるアプリケーション開発のためのライブ  
ラリの提案, 組込みシステムシンポジウム  
2016 論文集, pp.64-72
- 村山龍太郎, 谷沢智史, 西村一彦 (2015)  
Pepper プログラミング 基本動作からアプ  
リの企画・演出まで. SB クリエイティブ株  
式会社, 327p.
- Tanaka F., Isshiki K., Takahashi F., Uekusa M.,  
Sei R., Hayashi K. (2015) Pepper Learns  
Together with Children : Development of an  
Educational Application. *Proceedings of the  
15th IEEE-RAS International Conference on  
Humanoid Robots (Humanoids 2015)*, Seoul,  
Korea, November 2015, pp.270-275
- 渡邊裕 (2020) 幼児教育の現場における人型ロ  
ボットの活用 I —Pepper 教育アプリケーシ  
ヨン開発の試み—. 小池学園研究紀要, 18 :  
25-34

奥 恵 (埼玉東萌短期大学専任講師)  
渡邊 裕 (埼玉東萌短期大学准教授)

## 注

- 1) Pepper  
<https://www.softbank.jp/robot/>  
(参照日：2021.02.01)
- 2) ロボアプリ  
[https://www.softbankrobotics.com/jp/  
product/consumer/roboapp/](https://www.softbankrobotics.com/jp/product/consumer/roboapp/)  
(参照日：2021.02.01)
- 3) Choregraphe (コレグラフ) のインストール  
方法  
[https://www.softbank.jp/robot/consumer/  
support/trouble/data/choregraphe/](https://www.softbank.jp/robot/consumer/support/trouble/data/choregraphe/)  
(参照日：2021.02.01)
- 4) Pepper 社会貢献プログラム 2  
[https://www.softbankrobotics.com/jp/  
product/academy/](https://www.softbankrobotics.com/jp/product/academy/) (参照日：2021.02.01)