

ロボットの調理動作学習へ向けた取組み

濱園侑美[†] 小林一郎[†] 麻生英樹[‡] 長井隆行[§] 中村友昭[§] 持橋大地[¶]
[†]お茶の水女子大学 [‡]産業技術総合研究所 [§]電気通信大学 [¶]統計数理研究所

1 はじめに

家庭内にロボットが入ってきた際にロボットと居住者とが協調して暮らせる条件として、言葉だけでなく身振りなどを使って居住者の経験をロボットに伝え同じ動作を学習させることも必要になってくると考える。このことを踏まえ、本研究では、ロボットが人と同じ動作を真似て行うことができるように自らが行える基本動作を組み合わせてひとつの行為を生成する枠組みを分析し、複数の基本動作から複雑な行為を表現する対応関係を明確にすることを目的とする。また、対象とする行為を調理動作とし、基本動作からなる行為の具体的な関係について考察を行う。

2 ロボットの動作

2.1 ヒューマノイドロボットの概観

使用するロボットは(株)川田工業社製ヒューマノイドロボット HIRO を用いる。HIRO はオープンなソフトウェア環境を持つプラットフォームで、双腕、双眼を持ち、手元にもカメラがついている。図1にヒューマノイドロボットの概観を示す*。HIRO は首の左右上下運動に2つ、体全体を動かす1つ、両腕に各6つ、両手に各4つの関節を持つ。それぞれの関節角と時間 t を指定することで t 秒かけて指定された角度へと関節を動かすことが可能である。表1にそれぞれの関節が動く範囲を示す。

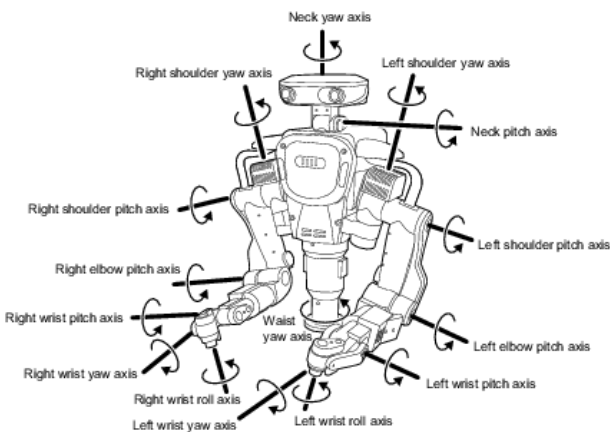


図1: ロボットの関節

表1: 関節の動作可能範囲

関節軸名称	記号	可動範囲 (deg)	最大速度 (deg/s)
腰ヨー軸	CY	-163 to +163	130
首ヨー軸	NY	-70 to +70	150
首ピッチ軸	NP	-20 to +70	300
右肩ヨー軸	RSY	-88 to +88	172
右肩ピッチ軸	RSP	-140 to +60	133
右肘ピッチ軸	REP	-158 to +0	229
右手首ヨー軸	RWY	-165 to +105	300
右手首ピッチ軸	RWP	-100 to +100	223
右手首ロール軸	RWR	-163 to +163	300
左肩ヨー軸	LSY	-88 to +88	172
左肩ピッチ軸	LSP	-140 to +60	133
左肘ピッチ軸	LEP	-158 to +0	229
左手首ヨー軸	LWY	-105 to +165	300
左手首ピッチ軸	LWP	-100 to +100	223
左手首ロール軸	LWR	-163 to +163	300

2.2 動作構成

ロボットの調理動作を関節軸の基本動作から構成するために、Chengら [1] による Activity-Attribute Matrix(AAM) を参考にする。AAM は動作と動作に関連している意味属性を符号化したものであり、 M を Activity (活動)、 N を Attribute (属性) とし、各要素 $a_{ij} (i \in M, j \in N)$ において Attribute の Activity への含有関係について Activity i を構成するのに Attribute j が用いられている場合は 1, 用いられていない場合 j は 0 とする、 $M \times N$ 行列により動作を表現する。本研究では、Activity として調理動作を例に捉え、それに伴い Attribute として右手の指先を前後に動かす (x)、左右に動かす (y)、上下に動かす (z) 基本ベクトルを設定した。また、ロボットを実際に動かすにはそれぞれの関節角を指定する必要がある。そこで、Attribute の x, y, z それぞれに対して、係数となるベクトルとの内積をとることにより、ロボットの動作生成を可能にする。さらに Cheng らは Activity に含まれている Attribute を参照することにより、Activity の推定を行なうことを対象としていたため、基本動作に対する時系列性や速度等は考えていなかったが、本研究では Activity を生成する過程において、それぞれの Attribute の度合い、時系列性、速度等が重要となってくるため、 x, y, z の変動割合と時間 t を組み合わせた $[p_x, p_y, p_z, t]$ を時系列に n 個並べた $[[p_{x_1}, p_{y_1}, p_{z_1}, t_1], [p_{x_2}, p_{y_2}, p_{z_2}, t_2], \dots, [p_{x_n}, p_{y_n}, p_{z_n}, t_n]]$ を与えることにより動作の生成を可能にした。

Toward Cooking Motion Learning of a Humanoid Robot

Yumi Hamazono[†], Ichiro Kobayashi[†], Hideki Asoh[‡]
 Takayuki Nagai[§], Tomoaki Nakamura[§], Daichi Mochihashi[¶]
 *文献 [2] より引用。

2.3 動作生成

表 2 に AAM を改良した提案 Matrix を示す．以下，次章にて表 2 がどのように使用されるかについて具体例と共に示す．

表 2: 提案する改良 AAM の概要

速く切る	0	0	-5	1	0	2	5	1	0	2	5	1	
ゆっくり切る	0	0	-5	5	0	2	5	5	0	2	5	5	
粗く切る	0	0	-5	2	0	4	5	2	0	4	5	2	
細かく切る	0	0	-5	2	0	1	5	2	0	1	5	2	
ざっくり切る	0	0	-5	5	0	4	5	5	0	4	5	5	
	p_{x_1}	p_{y_1}	p_{z_1}	t_1	p_{x_2}	p_{y_2}	p_{z_2}	t_2	...	p_{x_n}	p_{y_n}	p_{z_n}	t_n

CY	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
NY	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
NP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RSY	0.1	1.8	0.7	0.1	1.8	0.7	0.1	1.8	0.7
RSP	-2.3	0.7	0.1	-2.3	0.7	0.1	-2.3	0.7	0.1
REP	2.1	-0.5	-2.7	2.1	-0.5	-2.7	2.1	-0.5	-2.7
RWY	0.0	0.0	0.1	0.0	0.0	0.1	0.0	0.0	0.1
RWP	0.2	0.2	2.7	0.2	0.2	2.7	0.2	0.2	2.7
RWR	0.0	-1.8	0.0	0.0	-1.8	0.0	0.0	-1.8	0.0
LSY	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LSP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LEP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LWY	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LWP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LWR	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RH1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RH2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RH3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RH4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LH1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LH2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LH3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LH4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

間 t によって動作の種類を生成できる．表 2 を見ると，複数の種類の「切る」という動作が， y と t の値によって差別化されていることがわかる．次に，「速く切る」という動作を見てみると， $[[p_{x_1}, p_{y_1}, p_{z_1}, t_1], [p_{x_2}, p_{y_2}, p_{z_2}, t_2], \dots, [p_{x_n}, p_{y_n}, p_{z_n}, t_n]]$ の値は $[[0, 0, -5, 1], [0, 2, 5, 1], \dots, [0, 2, 5, 1]]$ となっており，時間の経過と共にその動作を示す係数ベクトルが表示されていることがわかる．

3. 基本ベクトルと係数ベクトルの内積，移動先座標の取得および速度の追加

表 2 中，「速く切る」における最初の係数行列は， $[p_{x_1}, p_{y_1}, p_{z_1}, t_1] = [0, 0, -5, 1]$ であり，それと基本ベクトルの内積の結果に現在の座標を加え，さらに動作の速度 t_1 を加えた結果，以下ようになる．

$$[[[0.0, 0.0, 0.0], [-4.1, -0.4, -86.4, 14.7, -3.9, 3.2], [0.6, 0.1, -99.9, -15.2, 9.6, -3.2], [0.0, 0.0, 0.0, 0.0], [0.0, 0.0, 0.0, 0.0]], 1]$$

この動作生成を引き続き $[p_{x_2}, p_{y_2}, p_{z_2}, t_2] = [0, 2, 5, 1]$ にも適用した結果のロボットの動作を図 2 に示す．

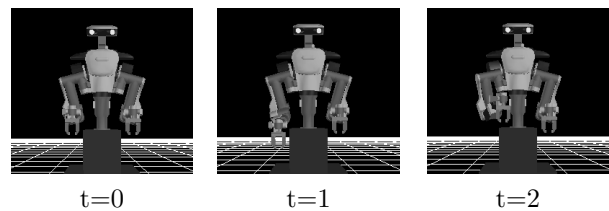


図 2: 「速く切る」の動作例

3 動作生成実験

3.1 動作生成手順

ロボットの動作生成の手順を以下に示す．

1. 基本ベクトルの準備：要素動作となる x, y, z 方向の基本ベクトルを準備する
2. 係数ベクトルの準備：1. で準備したベクトル方向にどの程度移動させるかを示すベクトル（係数ベクトル）を生成する
3. 基本ベクトルと係数ベクトルの内積：1. の基本ベクトルと 2. で生成された係数ベクトルの内積を計算し，関節角座標（以下「座標」と呼ぶ）を求める
4. 移動先座標の取得：3. で計算された座標に現在の座標を加えて，移動先の座標を取得する．ロボットに与える座標のフォーマットは以下の様になる．

$[[CY, NY, NP], [LSY, LSP, LEP, LWY, LWP, LWR]$
 $[RSY, RSP, REP, RWY, RWP, RWR],$
 $[RH1, RH2, RH3, RH4], [LH1, LH2, LH3, LH4]]$

具体例を上記順に従って以下に示す．

1. 基本ベクトルの準備
準備される基本ベクトルは，表 2 の x, y, z に示される値となる．
2. 係数ベクトルの生成
係数行列の生成を調理動作「切る」を例に説明する，切る動作は特に y (左右の動き) と時

3.2 実験結果と考察

動作生成に対応するため，Cheng らによる Activity-Attribute Matrix を参考にして動作要素の時系列変化を捉える関係行列を作成した．作成した行列を用い，シミュレータを利用してロボットが意図する動作を行っていることを確認した．

4 おわりに

本研究では，ロボットの動作生成を行うため調理動作生成を例に基本動作から対象となる動作を生成するための枠組みを提案し，シミュレータを使って提案手法の妥当性を検証した．今後の課題としては，より多種多様な動作生成を可能にするつもりである．

参考文献

[1] Heng-Tze Cheng, Feng-Tso Sun, Martin Griss, Paul Davis, Jianguo Li, Di You, “NuActiv: Recognizing Unseen New Activities Using Semantic Attribute-Based Learning”, MobiSys’13, Taipei, 2013.
 [2] <http://robot-support.kawada.jp/support/hiro/>