

ロボット喫茶店における飲み終わりを考慮した 給仕タイミングの検討

Discussion of Serving Model Considered The End of Drinking at Robot Cafe

中島 雅貴*¹
Masaki Nakashima

小篠 裕子*¹
Yuko Ozasa

萬 礼応*¹
Ayanori Yorozu

森田 武史*¹
Takeshi Morita

高橋 正樹*¹
Masaki Takahashi

斎藤 英雄*¹
Hideo Saito

*¹慶應義塾大学大学院 理工学研究科

Graduate School of Science and Technology, Keio University

Serving robot in cafe may take a cup when a customer finish drinking in a future. In order to understand the serving timing in this case, the robot should consider the end of drinking by customers. The serving timing is modeled by the score which is determined by the degree of drinking in this paper. When a customer is drinking, the score is updated so that the score becomes high.

1. はじめに

サービスロボットの進化に伴い、飲食店などで給仕に従事する給仕ロボットの開発が求められている [1]。多くの給仕ロボットは、予めプログラムされた給仕内容を行うのみであり、柔軟なサービスを提供しているとは言い難い。人の給仕のようなサービスを給仕ロボットに行わせるためには、人の様子や状況を考慮して、給仕内容を変えることが望ましい。

喫茶店など、飲み物を提供する飲食店では、飲み終わった飲み物を下げるといったサービスが行われている。お客さんが退店後に飲み物を下げるといった給仕も考えられるが、お客さんの飲み終わりを考慮して、飲み物を下げることが望ましい。本研究では、お客さんの飲み終わりを考慮して飲み物を下げるタスクを扱う。このようなタスク設定を扱った研究は我々の知る限りない。

本研究では、お客さんの飲み終わりを考慮して給仕タイミングをモデル化する方法を提案する。飲み物は、飲むたびに線形に減りゆくと仮定する。つまり、飲み物は飲む度に飲み終わりに近づく。ここで、飲み終わり具合をスコアとして定義することを考える。スコアが高ければ高いほど、飲み終わりに近づいているようにスコアを定義したい。このようにスコアを定義することができれば、スコアを閾値判定することで、給仕タイミングを定義することが可能となる。

提案手法では、人が飲んでいるか否かを識別し、飲んでいると判定されたときは、飲み終わり度としてのスコアが高くなるように、スコアを更新し、飲んでいないと判定されたときはスコアを更新しないという方法で、スコアを定義する。人が飲んでいるか否かの識別には、行動認識に広く使われている人の関節情報を特徴 [2] として、Nearest Neighbor (NN) [3] を識別器として用いる。飲み終わり度合を示すスコアが、ある閾値以上になれば、ロボットは飲み終わったと判定し、飲み物を下げにいけばよい。実験では、実際に人が飲み物を飲む様子を撮影し、撮影したデータを用いて、提案手法を評価する。

2. 飲み終わり度合スコアの算出

本研究では、飲み終わりに近づく度合をスコア S として表現する。飲み物は、人が飲む度に線形に減りゆくと仮定する

と、 S は、人が飲んでいると判定されたときに、値が大きくなるように更新すればよい。本稿では、人の関節情報を用いて、人が飲んでいるか否かの識別を行い、その結果を用いて S を更新する。

人の関節情報は、Microsoft 社製 Kinect v2 [4] から得る。Kinect v2 は人ひとりにつき、25 個の関節を取得可能である。人は、飲食店で飲み物を飲む際、着席していることがほとんどであることより、本研究では、上半身の 15 個の関節のみを用いる。ここで、 t フレーム目の、上半身 15 個の関節を $\mathbf{J}_t = \{j_{x_0}^t, j_{y_0}^t, j_{z_0}^t, \dots, j_{x_{14}}^t, j_{y_{14}}^t, j_{z_{14}}^t\}$ と定義する。ここで、 $(j_{x_i}^t, j_{y_i}^t, j_{z_i}^t)$ は、 t フレーム目の、 i 番目の関節の 3 次元座標を示している。

飲んでいるか否かの識別器には、NN を用いる。NN は、テストデータに最も近い学習データを探索し、その学習データのクラスに分類する方法である。 \mathbf{J}_t を特徴ベクトルとして、フレーム毎に飲んでいるか否かの 2 クラス識別をする。 t フレーム目の 2 クラス識別結果を r^t とする。飲んでいると判定されたら、 $r^t = 1$ 、飲んでいないと判定されたら、 $r^t = 0$ とする。

フレーム毎に得られた r^t を用いて、スコア S を表現する。連続する N フレーム中で飲んでいると判定 ($r^t = 1$) されたフレーム数が全体の $M\%$ である場合にスコアを更新する。ここで、 T を 1 セットにおけるフレーム数とする。スコアを更新するアルゴリズムを下記に示す。

Algorithm 1 飲み終わり度合スコア S 更新アルゴリズム

```

for  $t = 0$  to  $T - 1$  do
   $q \leftarrow t \bmod N$ 
   $tmp[q] \leftarrow r^t$ 
  if  $q = 0$  and  $\text{mean}(tmp) > M$  then
     $S \leftarrow S + 1$ 
  end if
end for

```

スコア S は、飲んでいると判定される度に上昇し、飲み終わった後はスコアは変化しなくなる。本研究では、スコアが上昇後、変化せずに F フレーム経過し、かつ、スコアが σ 以上になるタイミングを、ロボットが給仕行動を行うのに最適なタイミングであると定義する。

連絡先: 慶應義塾大学大学院理工学研究科,
〒 223-8522 神奈川県横浜市港北区日吉 3-14-1



図 1: 実験の様子

表 1: 関節情報を用いた飲用行動識別結果 (%)

	Accuracy	Precision	Recall	Specificity
A	98.0	81.4	85.0	98.8
B	96.3	64.2	60.4	98.1

3. 実験

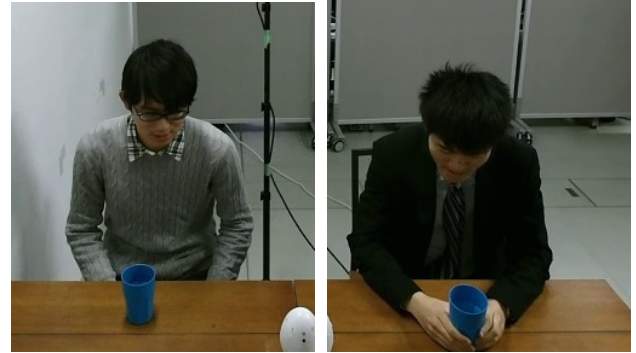
実験にあたって、人が飲み物を飲んでいる様子を撮影したデータセットを構築した。協力者 2 名に、コップ一杯分の飲み物を雑談を交えながら、飲んでいただき、飲み終わるまでを 1 セットとして撮影した。Kinect v2 によって、関節情報を計 3 セット取得した。図 1 は、実験の様子である。Kinect から見て左側の協力者を「協力者 A」、右側の協力者を「協力者 B」とする。

本研究の飲用状態の定義は、口にコップをつけている間とし、全データに対し、「飲んでいる」、「飲んでいない」のアノテーションを行った。図 2 は協力者 A, B の「飲んでいない」とアノテーションされた画像例、図 3 は「飲んでいる」とアノテーションされた画像例である。

本研究では、関節位置を用いた飲用行動の認識を行い、飲んでいるか飲んでいないかを識別し、飲み終わり度合いを示すスコアの変化について検証した。3 セットのうち、1, 2 セット目を学習データ、3 セット目をテストデータとし、学習データは 134940 フレーム分、テストデータは 7954 フレーム分が得られた。テストデータの飲用行動識別結果を表 1 に示す。「協力者 A」と「協力者 B」の正解率 (Accuracy)、適合率 (Precision)、再現率 (Recall)、特異度 (Specificity) を算出した。両者とも特異度が 98% を超え、本来は飲んでいないフレームを飲んでいると判定したフレームは少なかった。再現率は、「協力者 A」は 85.0% なのに対して、「協力者 B」は 60.4% となり、本来は飲んでいるフレームを飲んでいないと判定したフレームの数が多くなった。

飲用行動識別結果を元に、横軸をフレーム番号 t 、縦軸をスコア S とした t - S グラフを求めた。図 4 ~ 図 7 に、提案手法と真値の t - S グラフを示す。真値は、飲んでいるか否かに関する真のアノテーションを用いて、スコア S を算出した結果である。つまり、この真値は、飲用行動認識が全て正解した場合の S の変化を示す。

図 4 は、「協力者 A」、 $N=10$, $M=50$ とした時の t - S グラフである。 $t=7500$ 付近で本来は飲んでいないフレームを飲んでいると誤認識したが、これを図 5 に示す $N=30$ とした時、 $t=7500$ 付近の誤認識による S の加算は行われず、真値のグラ



(a) 協力者 A

(b) 協力者 B

図 2: 「飲んでいない」とアノテーションされた例。



(a) 協力者 A

(b) 協力者 B

図 3: 「飲んでいる」とアノテーションされた例。

フに近い結果が得られた。一方、図 6 は「協力者 B」、 $N=10$, $M=50$ とした時の t - S グラフである。 $t=200$, と $t=5500$ 付近に同じく誤認識したが、図 7 の $N=30$ とした場合、誤認識による S の変化への影響が緩和された。以上より、協力者 A, B 共に、 $N=10$ よりも、 $N=30$ の方が、 S は真値に近い変化をしたことが分かる。

次に、給仕タイミングの設定方法について考察する。例として図 5 の場合を考える。この場合の飲み終わり時、 $S=14$ となることから、飲み終わり判定の閾値 $\sigma=12$ と設定すると、図 5 の協力者 A に対しては、 $S \geq 12$ となって以降、 $F=1000$ フレーム経過した後にロボットに給仕行動の指令値を送るようにすることが理想的である。他の結果についても同様に考えると、理想的な σ や F を推論することはできるが、各ケースにおいて理想的だと考える σ や F の値は異なる。最適な σ や F の設定方法を今後の予定とする。

4. まとめ

提案手法では、上半身の関節情報を用いて飲用行動の NN による認識を行い、その結果を用いて、飲み終わりに近づく度合いを表すスコア S を定義した。実験では、 S 変化の様子を真値と比較することで、飲用行動認識の誤認識による影響は、 S を更新するフレーム間隔 N を大きくした方が小さくなるという知見を得た。今後は、この S を用いて、最適な給仕タイミングをモデル化するために必要な、飲み終わり判定の閾値などのパラメータを、機械学習手法を用いて設定する手法を提案する予定である。

更に、本手法を基に、最適な給仕タイミングを提供する

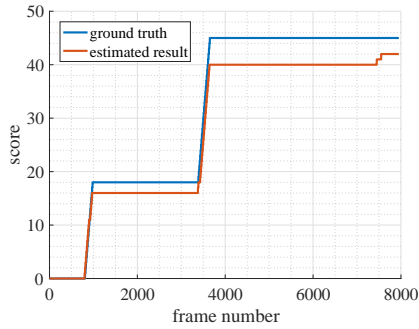


図 4: スコアの変化 (協力者 A, $N = 10, M = 50$)

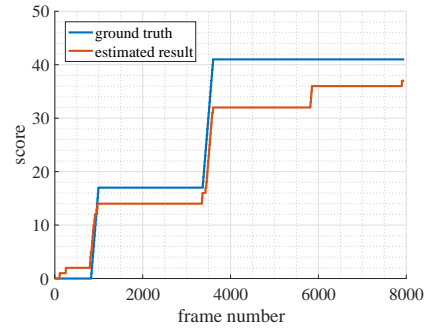


図 6: スコアの変化 (協力者 B, $N = 10, M = 50$)

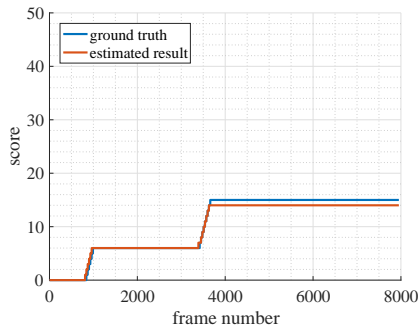


図 5: スコアの変化 (協力者 A, $N = 30, M = 50$)

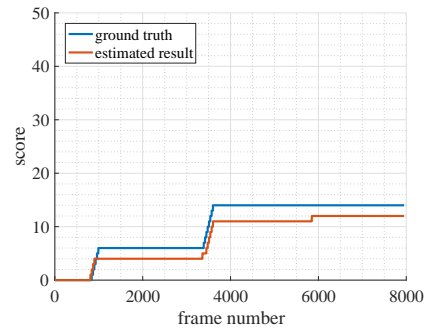


図 7: スコアの変化 (協力者 B, $N = 30, M = 50$)

PRINTEPS[5] モジュールを開発する予定である。PRINTEPS は、モジュールを視覚的に連結することで、容易に AI ロボットアプリケーションを開発可能とするワークフローエディタである。本手法を PRINTEPS モジュール化することにより、状況を理解しながら給仕をするようなロボットの開発を、誰もが容易にできる未来を導きたいと考えている。

謝辞

本研究は、科学技術振興機構 (JST) 戦略的想像研究推進事業 (CREST) 「実践知能アプリケーション構築フレームワーク PRINTEPS の開発と社会実践」の支援によって実施した。

参考文献

- [1] Yu, Qingxiao, et al. "An autonomous restaurant service robot with high positioning accuracy." *Industrial Robot: An International Journal* 39.3 (2012): 271-281.
- [2] Du, Yong, Wei Wang, and Liang Wang. "Hierarchical recurrent neural network for skeleton based action recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. (2015).
- [3] Cover, Thomas, and Peter Hart. "Nearest neighbor pattern classification." *IEEE transactions on information theory* 13.1 (1967): 21-27.
- [4] LaBelle, Kathryn. "Evaluation of Kinect joint tracking for clinical and in-home stroke rehabilitation tools." Undergraduate Thesis, University of Notre Dame (2011).
- [5] 実践知能アプリケーション構築フレームワーク PRINTEPS, <http://printeps.org/>