

リカレントニューラルネットワークと手の動きに特化した

ナチュラルユーザーインターフェースを用いたコンピュータ操作手法

A Computer Interface Realized by a Recurrent Neural Network and a Natural User Interface Based on Tracking Hand Motion

○ 楊嘉晨 (芝浦工大) 堀江亮太 (芝浦工大)

Jiachen YANG, Shibaura Institute of Technology
Ryota HORIE, Shibaura Institute of Technology

Abstract: We developed an interface system by which a user can operate a computer by movement of the hand and fingers, without touching with a keyboard and a mouse. To implement the interface, we used a gesture sensor (Leap Motion Controller, Leap Motion, Inc.) to acquire data for movement of the hand and fingers. We used Recurrent Neural Network (RNN) to discriminate types of gestures from the data. In our proposed system, the user can execute shortcut keys of a computer by making the gestures. The user can also use functions of a mouse by moving the hand and fingers. In an experiment, three types of gestures were discriminated by using the RNN. RNN outputted values corresponding to the gestures successfully. The outputs were transferred to their corresponding shortcut keys of the computer. As an application of our proposed system, we could control household appliances by using ECHONET Lite.

Key Words: Recurrent Neural Network, Natural User Interface, Leap Motion

1. 背景

標準的なコンピュータの入力インターフェースはキーボードとマウスである。ユーザーはマウスとキーボードを組み合わせることでコンピュータの操作を行うが、常に手でキーボードやマウスに触れている必要がある。いっぽう近年、手や指の動きが測定できるセンサーが普及してきている。そこで、本研究ではキーボードとマウスに触れずに手の動きを用いてコンピュータを操作するインターフェースの実現方法を考案した。手の位置が限定された状況でも使用できる生活支援技術のシーズ技術として提案したい。

2. 目的

本研究ではユーザーの手の動きに対応してマウスの移動機能と操作機能を実現するとともに、ジェスチャーの種類でコンピュータのショートカットキーを実行するインターフェースを提案した。提案システムでは、ユーザーの手の中心座標の時間発展および指の動作パターンをジェスチャーセンサー (Leap Motion 製 Leap Motion) で読み取る。手のジェスチャーに伴う手の中心座標の時系列をリカレントニューラルネットワーク (RNN) ⁽¹⁾ に入力してジェスチャーの種類を判別を行い、コンピュータのショートカットキーに対応させる。

3. 方法

本研究で提案するシステム構成を Fig. 1 に示す。本研究では Visual Studio Express 2012 for Windows Desktop C++ および LeapDeveloperKit2.0.2.16391Beta for Windows を用いてシステムの開発をした。

ユーザーが手を閉じてから開けるまで、手のジェスチャーに伴う手の中心の三次元座標の時系列が Leap Motion によって測定される。三次元座標はジェスチャーが行われる場所とジェスチャーの大きさによって異なるため、前後二つの時刻の座標の差分ベクトルを導出し、単位ベクトルに変換することでジェスチャーの場所と大きさによる影響を軽減することができる。この単位ベクトルの時系列を RNN

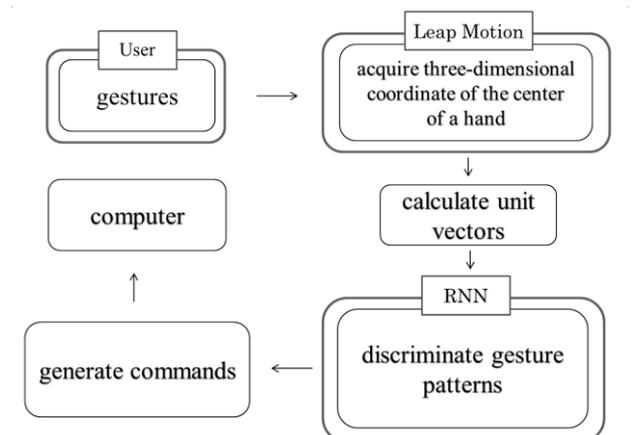


Fig. 1 System configuration

の入力とした。RNN は時系列データを入力として扱え、入力に対応する出力時系列データを学習することができる。RNN には3種類のジェスチャーをそれぞれ異なる時系列データが出力するように学習させた。RNN の学習時間は代入されるデータの大きさと時系列長によって左右される。データを上記の単位ベクトルに変換し、ダウンサンプリングして RNN に代入することでデータの大きさと時系列長を抑え、RNN の学習時間を減少させることができる。本研究では約 60 サンプル点のデータを 10 サンプル点にダウンサンプリングした。RNN の出力に対応するショートカットキーが実行される。以上より、ユーザーは目的に応じてジェスチャーを行い、ショートカットキーを実行することができる。

提案システムでは、以上とは別の直感的な入力機能を実行することができる。ユーザーの手の中心座標の動きに対応してマウスの移動操作が可能になる。また、幾つかの指部の動きが Leap Motion によって検出されると、マウスのボタン操作が実行される。以上の手の中心座標の動きと指

部の動きに対応したマウスの機能を Table 1 に示す. これらは Leap Motion の API 関数^(2,3)を主に用いて開発した.

Table 1 Functions of a mouse corresponding to movement of the hand and fingers

movement of the hand and fingers	Functions of a mouse
move the hand	move a mouse pointer
tap the forefinger	click a left mouse button
tap the middle finger	click a right mouse button
Pinch of a thumb and the forefinger	drag operation

4. 結果

3 種類のジェスチャーのデータをそれぞれ 1 つ用意し RNN に代入し学習を行う. 学習が終了したらそれぞれのジェスチャーに対して学習で使われたデータ以外の 100 個のデータを用意してシミュレーションを行う. シミュレーションの結果を Table 2 に示す. ほぼすべてのジェスチャーが正しく認識されたことが分かる.

Table 2 Recognition rate of the gestures

gestures	recognition rate
draw a straight lines from the left to the right	100%
draw a straight line to the depths	98%
draw a straight lines from the right to the left	100%

RNN の出力と教師信号である時系列データの誤差の推移を Fig. 2 に示す. 横軸は学習回数, 縦軸は誤差の大きさを表している. 誤差の大きさは最初の数回で急激に減少して, くだらかに収まっていくことが分かる. 誤差が大きいときに学習を終了すると RNN の出力は教師信号である時系列データと異なる時系列データを出力するため, 誤差が小さくなるまで学習を行う必要がある.

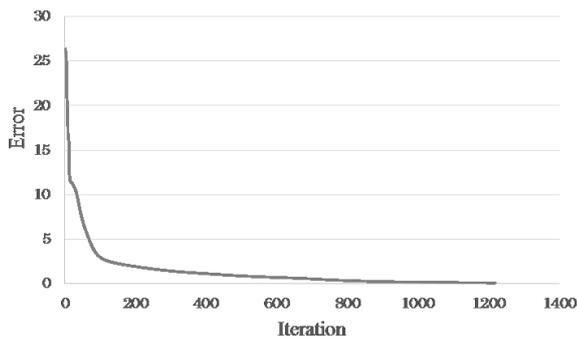


Fig. 2 Learning curve of RNN

手の中心座標の動きと指部の動きに対応したマウスの機能は正しく認識された. また, 提案システムを用いた上記の実験風景を Fig.3 に示す.

5. 考察

Table2 において, 奥への直線を描くジェスチャーが 100% 認識できなかった原因は, まれにジェスチャーを取得するタイミングがずれることである. 手を奥まで突き出してジ

ェスチャーを終了させた後に, 手前まで手を戻すときもジェスチャーとして認識されデータが取得されたため, 奥への直線を描く手の運動パターンとはならず, 誤認識が発生したと考えられる. ユーザーがジェスチャーを行う際は, ジェスチャーの開始と終了をシステムに認識させる手の開閉を注意深く行う必要があることが分かった.



Fig. 3 Experiment image

6. 結論

RNN と Leap Motion を用いて, ジェスチャーによりショートカットキー入力を, 手の動きによりマウスの機能を入力するインターフェースを開発した. 提案するインターフェースの応用として, ECHONET Lite⁽⁴⁾と組み合わせることでジェスチャーによる家電の操作を行うことができる. 詳細は今後に報告する. ジェスチャーによるコンピュータ等の操作をより行いやすくすることが今後の開発の課題である.

参考文献

- (1) Jun Tani, Naohiro Fukumura, Embedding a Grammatical Description in Deterministic Chaos: An Experiment in Recurrent Neural Learning, Biological Cybernetics, vol. 72, no. 4, pp.365-370, 1995.
- (2) Leap Motion社, API Reference, https://developer.leapmotion.com/documentation/skeletal/cpp/api/Leap_Classes.html
- (3) 中村薫, C++で始めるLeap Motion開発, <http://www.buildinsider.net/small/leapmotioncpp>
- (4) 株式会社ソニーコンピュータサイエンス研究所, <http://kadecot.net/blog/1479/>, MoekadenRoom: エアコン・照明・カーテン・温度計のエミュレータ