

テレグジスタンスによる非同期コミュニケーションの実現

産業能率大学 情報マネジメント学部 川野邊研究室 4年 相川 智貴 指導教員:川野邊 誠

研究概要

研究背景

- 厳しい作業環境で人の代わりに機械が作業を行い、安全な場所にいる人が遠隔で操作可能な技術は、災害対応・遠隔医療など幅広い応用が期待されている
- 従来のテレグジスタンスの研究では視覚に比べて触覚の再現が不十分でありリアルタイムでの触覚フィードバックが重要な課題であると述べられている



- リアルタイムでの接触が不可能な状況においても記録された触覚情報が再現可能であれば新たな体験価値を提供することが可能
- 空間立体ディスプレイと触覚フィードバックを組み合わせることで視覚と触覚を用いる握手の形が実現可能と考えた

※テレグジスタンス:物理的に離れた場所にいる人や物を、まるでそこにいるかのように感じる技術

研究目的

テレグジスタンスと空間立体ディスプレイを組み合わせることによりリアルタイムでの接触を必要とせず、触覚の再現可能なインタラクションを実現するシステムの開発

期待効果

- 物理的な距離や時間を超えて触覚を通じた感情の伝達が可能
- 新たな遠隔コミュニケーションの形を提供可能

研究構想

加藤らの「テレグジスタンスの研究 -TELESAR VI ハンド指先のための力・加速度・温度センシングシステムの開発-」^[1]



握手時の力、振動、温度などの触覚情報を高精度に取得するための力、加速度、温度センサを備えた指先用デバイスを開発



収集した触覚データを解析してユーザにフィードバックする触覚提示デバイスを設計、開発



空間立体ディスプレイに統合するための専用インターフェースを設計、実装

図1. 研究フロー図

研究成果

加速度、圧力、温度センサを備えた指先用デバイスの開発

使用機器

- Raspberry Pi5
- 9軸加速度センサ:BN0055
- 圧力センサ:FSR402
- A/Dコンバータ:ADS1015
- 温度センサ:SHT31D

開発環境

- Linux
- Ubuntu
 - Linuxのディストリビューション

使用ライブラリ

- Adafruit-blinka:i2cを制御
- Adafruit-circuitpython-bno055:加速度センサを制御
- Adafruit-circuitpython-fsr402:圧力センサを制御
- Adafruit-circuitpython-ads1x15:A/Dコンバータを制御
- Adafruit-circuitpython-sht31d:温度センサを制御

指先用デバイス構想

- 加速度、圧力、温度センサをそれぞれ指先に配置
 - 加速度センサは指の頂点で一番移動する場所
 - 圧力、温度センサは人に触れる部分
- 取得したデータはCSVデータとして保存

データ出力

- 9軸加速度センサ
 - 加速度、角速度のデータを取得
- 圧力センサ
 - 電圧のデータを取得
- 温度センサ
 - 表面温度のデータを取得

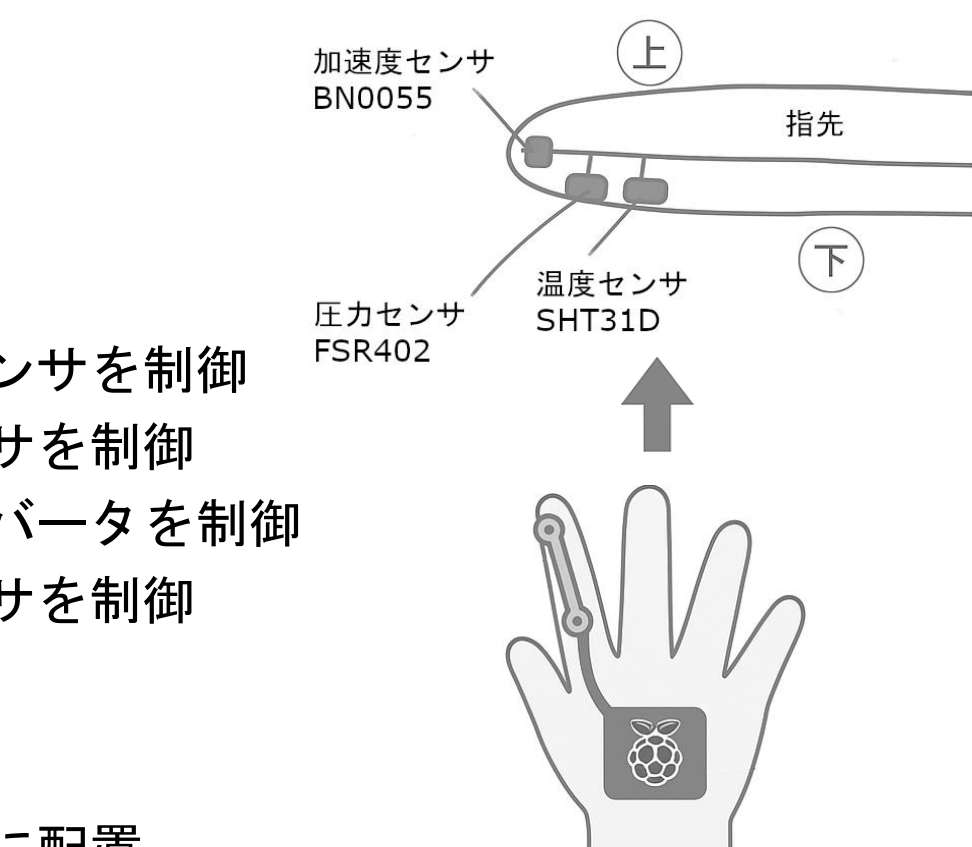


図2. 指先用デバイスイメージ図

```
a3225261@raspberrypi1:~$ python3 9jiku.py
BeginLog,stop Ctrl+C
Time: 0.00s Accel:(0.0, 0.0, 0.0) | Gyro:(0.0, 0.0, 0.0)
Time: 0.50s Accel:(-2.12, 5.89, 6.69) | Gyro:(-0.81399, -0.96199, -0.80218)
Time: 1.01s Accel:(-3.41, 5.0, 6.34) | Gyro:(-0.18217, -0.12654, 0.44254)
Time: 1.51s Accel:(-6.49, 4.41, 1.18) | Gyro:(-1.15715, 0.51497, -0.89281)
Time: 2.01s Accel:(2.68, -1.34, 9.94) | Gyro:(-0.42186, 1.69951, 0.00436)
Time: 2.52s Accel:(0.6, 4.58, 2.09) | Gyro:(2.76347, 2.40201, -3.15141)
Time: 3.02s Accel:(-2.12, 4.75, 2.53) | Gyro:(-2.42274, -0.71559, 0.15163)
Time: 3.52s Accel:(2.89, 0.93, 5.31) | Gyro:(0.28889, -2.01476, 0.36586)
Time: 4.02s Accel:(-4.15, 5.76, 2.73) | Gyro:(1.94855, 2.8884, -2.65200)
Time: 4.53s Accel:(-4.91, 5.05, 6.55) | Gyro:(4.89511, -3.07614, 2.15984)
Time: 5.04s Accel:(5.2, 0.68, 10.85) | Gyro:(0.29015, -1.44485, -0.17080)
```

図3. 加速度センサのデータ

```
a3225261@raspberrypi1:~$ python3 atunyoku.py
Time: 0.00s Raw: -1584 Voltage: 0.000 V
Time: 0.50s Raw: 0 Voltage: 0.000 V
Time: 1.01s Raw: -16 Voltage: -0.002 V
Time: 1.51s Raw: -16 Voltage: 0.000 V
Time: 2.02s Raw: 32725 Voltage: 4.094 V
Time: 2.52s Raw: 32725 Voltage: 4.094 V
Time: 3.03s Raw: 32752 Voltage: 4.094 V
```

図4. 圧力センサのデータ

```
a3225261@raspberrypi1:~$ python3 temperature.py
Time: 0.00 s Temperature: 26.34 °C
Time: 1.02 s Temperature: 26.58 °C
Time: 2.03 s Temperature: 26.76 °C
Time: 3.05 s Temperature: 27.88 °C
Time: 4.07 s Temperature: 28.74 °C
Time: 5.08 s Temperature: 29.41 °C
```

図5. 温度センサのデータ

結果

- センサを動かした際に数値が可動
- 時間、加速度、圧力、温度データをCSVデータとして保存完了
- 経過時間と他のセンサを対応付けることが可能

触覚提示デバイスの開発

触覚提示デバイスの概要

- 指先用デバイスにて保存した握手のデータを用いて、触覚提示デバイスに統合し、触覚を再現

使用機器

- Raspberry Pi5
- エレキットサイボーグハンド^[2]
- 高トルクサーボモータ:DS3218
- サーボモータ拡張モジュール:PCA9685

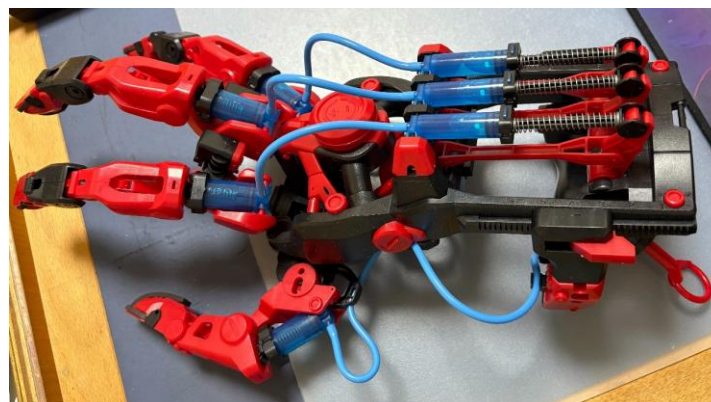


図6. エレキットサイボーグハンド^[2]

使用ライブラリ

- Adafruit-blinka:i2cを制御
- Adafruit-circuitpython-pca9685:高トルクサーボモータを制御
- Adafruit-circuitpython-busdevice:i2Cの通信層を制御

触覚提示デバイスの構想

- エレキットサイボーグハンドのバネ部分へモータ制御改造
- モータ制御により、握手データ以上の押し込みを制御
- モータ制御は糸または棒でバネ部分を引く
- バネは戻りの役割を持ち、戻りの強さを調節可能
- 温度はペルチェ素子を使用し、手の温度を再現

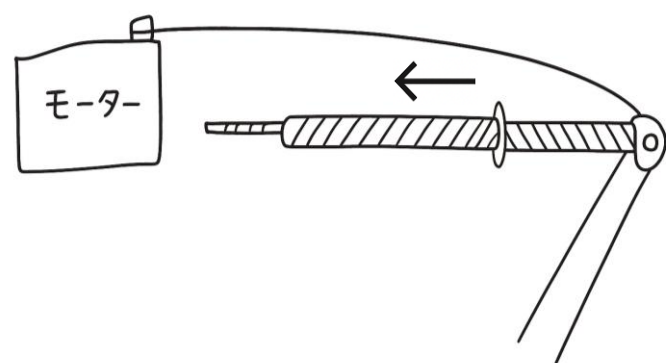


図7. バネ部分の改造イメージ図

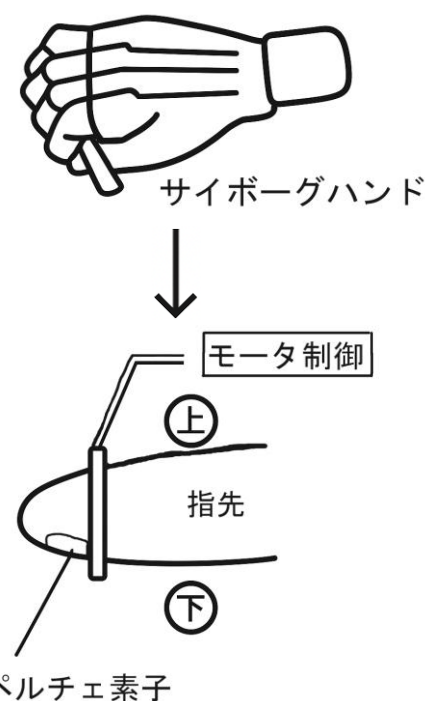


図8. 触覚提示デバイスイメージ図

結果

- 高トルクサーボモータを指定した角度に動かすことが可能
- サーボモータ拡張モジュールに2つサーボモータを繋ぐと、電力不足で稼働不可
- 握手データを取り込むことにより握手再現可能

今後の予定

- バネ部分を糸や棒で引く動作の検証
- 実際に使用可能か検証

参考文献

- [1] 加藤史洋, 井上康之, 舘暲, TELESAR VI ハンド指先のための力・加速度・温度センシングシステムの開発, 第15回テレグジスタンス研究会, 2020.
- [2] ELEKIT. サイボーグハンド [MR-9112]
<https://www.elekit.co.jp/product/MR-9112>, 2024.