広島工業大学紀要研究編
第59巻（2025）69-78
論　　文

Sigfoxを用いたIoT機器の作製とデータ処理

田中　武[1)]・渡部　雅也[1)]・古川　友樹[2)]・永山　和樹[2)]・飯間　捺希[2)]・多田　覚[1)]
岩堀　祐子[1)]・山内　将行[1)]・角川　幸治[1)]

（令和6年11月8日受付）

Fabricating IoT devices using Sigfox and processing data

Takeshi TANAKA[1*)], Masaya WATANABE[1)], Yuki FURUKAWA[2)], Kazuki NAGAYAMA[2)], Natsuki IIMA[2)]
Satoru TADA[1)], Yuko IWAHORI[1)], Masayuki YAMAUCHI[1)], Koji KAKUGAWA[1)]
(Hiroshima Institute of Technology[1)*], DynaxT Co., Ltd.[2)])

(Received November 8, 2024)

Abstract

We are developing a system that uses Sigfox to measure the environmental data of living spaces, such as temperature, humidity, air pressure, and carbon dioxide concentration, 140 times a day.

Furthermore, we designed and manufactured a printed circuit board to integrate the sensor with an Arduino processor. In this study, we applied the system we have developed so far to agriculture. In particular, using optical, ultrasonic, and image processing, we investigated a system to measure the length of green onions grown in a greenhouse.

Key Words: carbon dioxide sensor, internet of things, sigfox, agriculture

1　はじめに

情報化社会（Society4.0）では知識や情報の交流が未だ進んでおらず、学際的な連携不足による課題が生じています。 Society5.0では、あらゆる人やモノがIoT（Internet of Things）でつながり、さまざまな知識や情報が共有されます。

このような背景から、私たちはSigfoxネットワークとインターネットを利用して、温度、湿度、気圧などのセンサーを搭載し、さまざまな物理データを計測するIoTデバイスを開発しました[1-5]。

ここでは、教育・研究に応用するために、各種センサーを用いて、ネギの高さを測定したので報告する。

2．ねぎの高さの検討（赤外線モジュール（フォトリフレクター）によるネギ高さ測定の検討）

コントローラ：ESP32
外部通信：Sigfox（データはAPIでgoogleスプレッドシートに格納）
測定項目：反応の有無（0/1の信号）

採用したセンサー
赤外線センサモジュール：型式不明（amazonで10個入り999円）

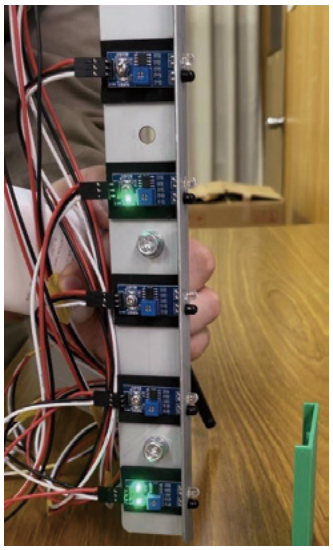


Fig.1　赤外線モジュールを50mm の等間隔で配置し、ネギに近接し測定

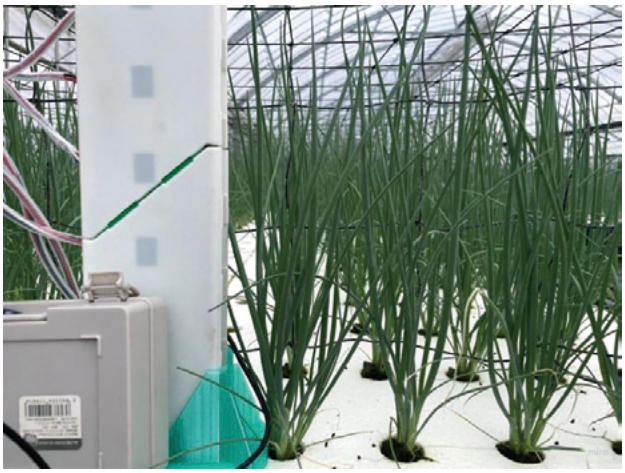


Fig.2　赤外線モジュールを50mm の等間隔で配置し、ネギに近接し測定

Fig.3　日中の太陽光による外乱を考慮し、照度センサーで200lux以下であるときにのみ測定するようプログラムした

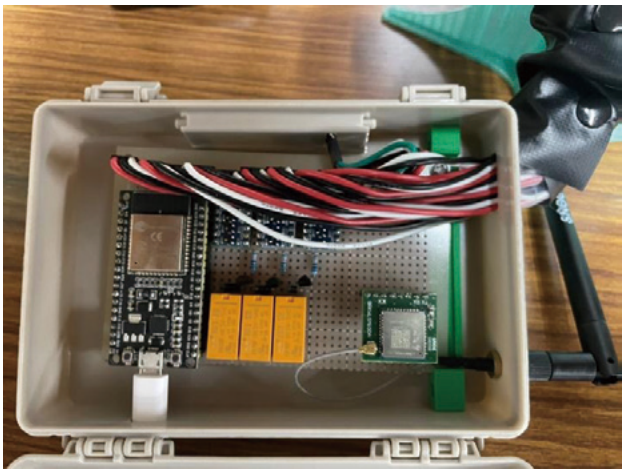


Fig.4　隣接する赤外線モジュールによる誤作動を防ぐため、リレー回路を組み時間差で駆動するようにした

測定対象：

・測定が容易な場所から任意の個体を選別し、葉身をスケールで計測し TOP 3 の値の平均値をスプレッドシートへ手動で記録した

・TOP 3 の葉身の太さもデジタルノギスで計測し、TOP 3 の太さの平均値も記載した

測定方法

・①ネギの株に可能な限り近づけ測定　②少しずらして再度測定し数回　①～②を数回繰り返し測定　③ネギの株から 5 cm 離して測定

・測定データは Sigfox Cloud の callback 機能を使用し、API で google スプレッドシートへ格納し、その結果を Fig.5に示す。

実測長平均(mm)	実測径平均(mm)	反応点(mm)	計測距離(data	device	seqNumber
294	5	300		10/11/2024 17:56:57 30CE870	149
294	5	100		10/11/2024 17:59:04 30CE870	150
294	5	100	50	10/11/2024 18:00:27 30CE870	151
294	5	150	50	10/11/2024 18:02:02 30CE870	152
294	5	200	50	10/11/2024 18:03:37 30CE870	153
240	3	200	0	10/11/2024 18:12:04 30CE870	154
240	3	150	0	10/11/2024 18:14:12 30CE870	155
240	3	150	50	10/11/2024 18:14:25 30CE870	156
240	3	150	50	10/11/2024 18:15:02 30CE870	157
240	3	150	50	10/11/2024 18:16:27 30CE870	158
200	2	100	0	10/11/2024 18:24:44 30CE870	159
200	2	100	0	10/11/2024 18:26:02 30CE870	160
145	1.5	100	0	10/11/2024 18:37:27 30CE870	161
145	1.5	100	0	10/11/2024 18:38:06 30CE870	162
115	1	50	0	10/11/2024 18:43:59 30CE870	163
115	1	50	0	10/11/2024 18:44:14 30CE870	164
115	1	50	0	10/11/2024 18:44:42 30CE870	165
350	4	200	0	10/11/2024 19:14:05 30CE870	166
350	4	200	0	10/11/2024 19:14:31 30CE870	167
350	4	200	50	10/11/2024 19:15:19 30CE870	168
350	4	200	50	10/11/2024 19:15:43 30CE870	169
310	4	250	0	10/11/2024 19:23:54 30CE870	171
310	4	200	0	10/11/2024 19:24:19 30CE870	172
310	4	200	50	10/11/2024 19:25:17 30CE870	173
310	4	150	50	10/11/2024 19:25:47 30CE870	174
310	4	100	50	10/11/2024 19:26:01 30CE870	175
350	4	200	0	10/11/2024 19:29:28 30CE870	176
400	6	350	0	10/11/2024 19:36:43 30CE870	177
400	6	350	0	10/11/2024 19:36:57 30CE870	178
400	6	250	0	10/11/2024 19:37:13 30CE870	179
400	6	350	0	10/11/2024 19:37:28 30CE870	180
400	6	200	0	10/11/2024 19:37:41 30CE870	181
400	6	200	0	10/11/2024 19:37:56 30CE870	182
400	6	300	0	10/11/2024 19:38:11 30CE870	183
400	6	100	0	10/11/2024 19:38:25 30CE870	184
400	6	200	50	10/11/2024 19:39:43 30CE870	185
400	6	200	50	10/11/2024 19:39:58 30CE870	186
400	6	100	50	10/11/2024 19:40:12 30CE870	187
445	8	200	0	10/11/2024 19:54:01 30CE870	189
445	8	200	0	10/11/2024 19:54:15 30CE870	190
445	8	200	0	10/11/2024 19:55:06 30CE870	191
445	8	200	0	10/11/2024 19:55:38 30CE870	193
445	8	300	0	10/11/2024 19:55:49 30CE870	194
445	8	200	50	10/11/2024 20:20:48 30CE870	195
445	8	200	50	10/11/2024 20:21:10 30CE870	196

Fig.5　測定結果

測定結果のグラフを、Fig.6に示す。

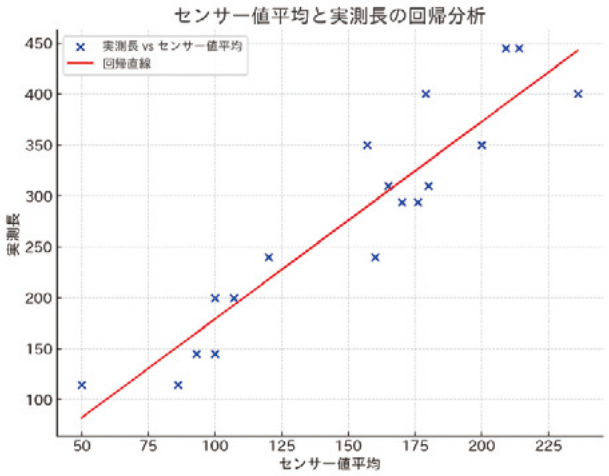


Fig.6　ねぎの長さにおけるセンサー値平均と実測長さの回帰分析

ネギ先端部の測定はできておらず、正確な測定には至らなかったが、得られた測定値の平均を算出し、ネギの実測長との回帰分析を行ってみた。

回帰式は以下の通り

実測長 =1.938×センサー値平均－14.53

であり、実測長を推測できる可能性を示した。さらに、多くの実測長と測定値の平均値データを分析することで収穫時期の判定の可能性が示された。

今後は、別のセンサーを用いて測定する予定である。

3．音波距離センサーによるネギ高さ測定の検討

3. 1　測定装置の概要

コントローラ：Arduino UNO R 4

外部通信：IDE にて PC に記録

測定項目：距離（cm）

採用したセンサー

超音波距離センサー：HC-SR04（OSEPP 製）[6)]

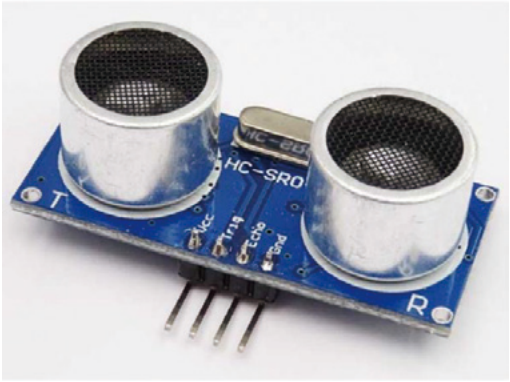


Fig.7　超音波距離センサー：HC-SR04（OSEPP 製）の写真

この超音波距離センサー モジュール（HC-SR04、Fig.7参照）は、ロボット製作や測定システムでよく使用されるコンポーネントです。最大400cm の非接触測定が可能です。DIY エレクトロニクス プロジェクトに最適なコンポーネントです。

HC-SR04の機能

動作電圧： 5 V

動作電流：15mA

距離範囲： 2 cm～400cm

100%Arduino 互換

3. 2　超音波を用いた測定方法

成長しているネギに対して、超音波を照射し、距離を測定しました。

ネギの横から、5 ㎝離して測定中の写真を Fig.8に示す。ネギの上から、土台の白いパネルから60㎝の位置で測定中の写真を Fig.9に示す。

Fig.8　ネギの横から５㎝離して超音波を照射して測定している写真

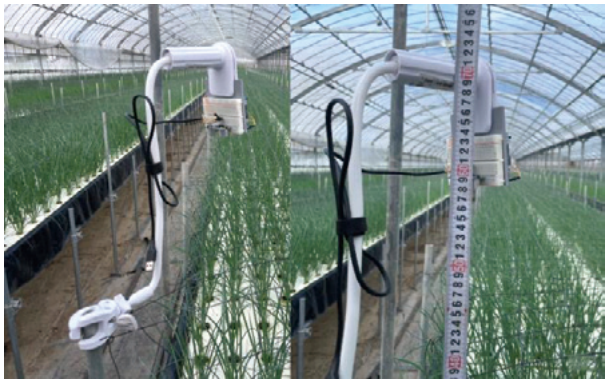


Fig.9　ネギを真上から超音波を照射し測定している写真。

測定対象および測定内容についてまとめると、:

・測定が容易な場所から任意で個体を選別し、葉身をスケールで計測し TOP 3 の値の平均値をログの名称とした。
・TOP 3 の葉身の太さもデジタルノギスで計測し、TOP 3 の太さの平均値も記載した。
・100mm 秒間隔で100回測定を行い、測定された距離の最大値・最小値・平均値・中央値・標準偏差を計算し記録した。
・①の測定（ネギの横から測定（Fig.8参照）は先端部から根っこにむけて順次測定し、センサーに安定的な反応があった箇所を測定点とした。
・②の測定（ネギの上から測定（Fig.9参照）は水耕栽培パネルから60cm の高さで固定し、それぞれの群を測定した。

3. 2. 1　ネギの横から超音波を照射した場合の結果

	A	B	C	D	E	F	G	H
1	葉身長TOP3平均値(mm)	葉身径(太さ/mm)	最大値(cm)	最小値(cm)	平均値(cm)	中央値(cm)	標準偏差(cm)	測定可能高さ(cm)
2	90	0.9	測定不可	測定不可	測定不可	測定不可	測定不可	測定不可
3	117	1	測定不可	測定不可	測定不可	測定不可	測定不可	測定不可
4	136	1.2	375	5	57.97	7.5	83.45	2
5	176	1.6	224	4	41.91	9	64.1	4
6	176	1.6	65	21	39.77	38	15.76	4
7	236	2	148	8	41.6	29.5	32.71	8
8	273	3	103	5	17.74	8	23.05	14
9	273	3	83	8	29.52	20	20.08	14
10	315	4.5	155	18	81.24	88	35.06	15
11	315	4.5	82	8	19.36	11	17.39	15
12	315	4.5	13	7	9.87	10	1.38	15
13	336	5.8	177	17	112.84	132	39.13	15
14	336	5.8	279	12	45.2	38	47.98	15
15	336	5.8	83	10	22.32	13	19.03	15
16	360	6	132	2	36.56	11	38.66	15
17	360	6	101	6	45.61	11.5	41.13	15
18	385	6	120	11	47.47	40	24.13	28
19	385	6	98	6	36.25	28	27.78	28
20	390	6	104	5	19.43	16	14.01	28
21	473	6.5	385	6	54.1	39.5	45.38	30
22	473	6.5	101	7	31.7	15	31.21	30

Fig.10　ネギの横から超音波を照射した場合の結果

植え付け早期段階のものは、超音波のほとんどがすり抜けていくため、測定不可能であった。

ある程度成長したものは測定可能であるが、先端部が疎な空間のため、全長の半分程度の場所で測定できた、その結果、測定可能高さを用いて、ネギの収穫時期を判定できる可能性がある。

今後は、ネギの収穫時期に、測定点の高さをパラメータとして、ネギに対する測定回数を増加させることにより、収穫時期を推定できる可能性が示唆された。

3. 2. 2　ネギの上から超音波を照射した場合の結果

葉身長TOP3平均値(mm)	葉身径	最大値(cm)	最小値(cm)	平均値(cm)	中央値(cm)
		60	58	58.6	59
		60	57	58.35	58
		59	57	58.19	58
		61	57	58.28	58
		64	46	53.78	54
		52	38	41.13	40
		44	40	41.28	41
		129	43	51.69	48.5
		70	33	52.94	54
		33	30	31.44	31
		91	10	32.19	32
		110	11	47.66	55
		114	44	55.81	53
		116	19	56.45	39
		37	23	25.87	25
		116	23	68.66	60
		29	22	25.42	26
		233	7	58.83	31

Fig.11　ネギの上から超音波を照射した場合の結果

植え付け早期段階のものは、超音波のほとんどが水耕栽培パネルからの反射によりパネルまでの距離が正確に測定できた。

ある程度成長した群のデータを俯瞰してみてみると、離散的な数値であるので、測定数を増加させることにより分析できる可能性が示唆された。

今後は、ネギの葉身の長さを記録し、大量の測定データを取得し、データ分析することにより、ネギの葉身の長さに関連する特徴を求めていきたいと思います。

4．カメラ画像の画像処理によるネギ高さ測定の検討

4.1　測定装置の概要

コントローラ：NVIDIA 製 Jetson Nano 2 GB の写真は Fig.12に、仕様は Table1に示す。
外部通信：USB メモリにデータ保存
測定項目：カメラ画像

採用したセンサー
モーションセンサー：Kinect v2（Kinect for Windows）（写真を Fig.13に示す。）

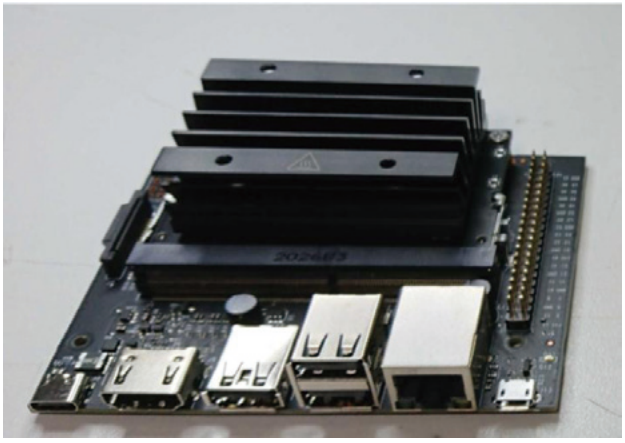


Fig.12　NVIDIA 製 Jetson Nano 2 GB の写真[7)]

Table1　NVIDIA 製 Jetson Nano 2 GB の仕様[7)]

GPU	128基のNVIDIA CUDAコアを実装したNVIDIA Maxwellアーキテクチャ
CPU	クアッドコア ARM Cortex -A57 MPCoreプロセッサ
メモリ	2GB 64ビット LPDDR4
ストレージ	microSD(別売)
ビデオエンコード	4K @30 (H.264/H.265)
ビデオデコーダー	4K @60 (H.264/H.265)
サイズ	100mm×80mm×29mm

Fig.14　Kinect v2（Kinect for Xbox One）の写真

4.2　ネギの画像データに画像処理を行い、ニューラルネットワークで学習

Step ①：画像に対して、緑色をフィルタリング
Step ②：Step ①の出力を 8 x 8 のグレースケールに変換
Step ③：Step ②のグレースケール画像を閾値150で白黒 2 値に変換（この64bit データを Sigfox で送信）
Step ④：Step ③のデータ（64bit/16進数）を 2 値化画像に戻し、その画像に対して 2 次元フーリエ変換（この画像に対し、教師データとしてラベリングし CNN で学習モデルを構築）

Fig.15　Step ①の写真

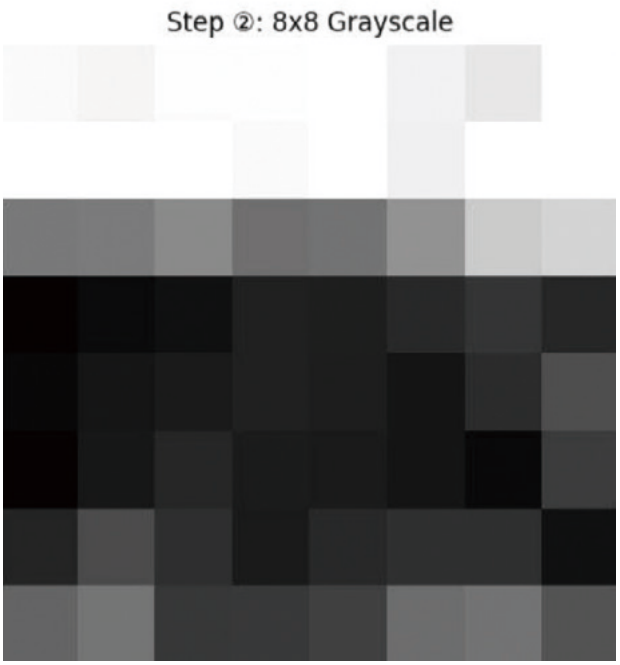


Fig.16　Step ②の写真

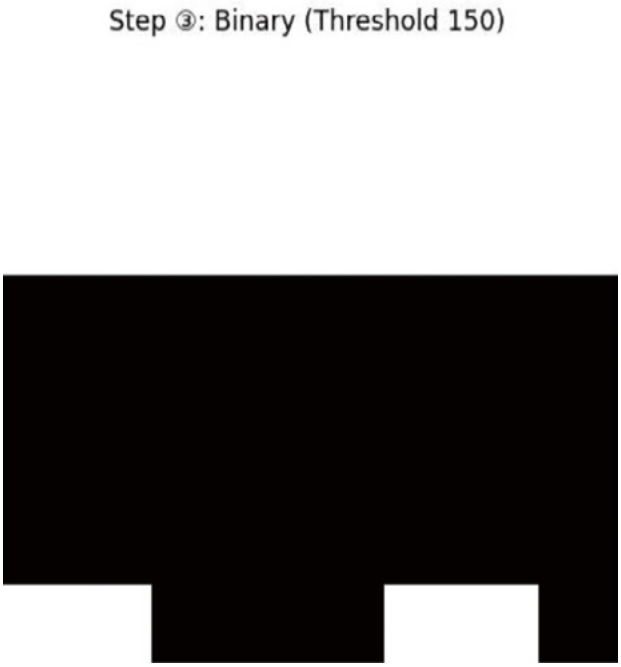


Fig.17　Step ③の写真

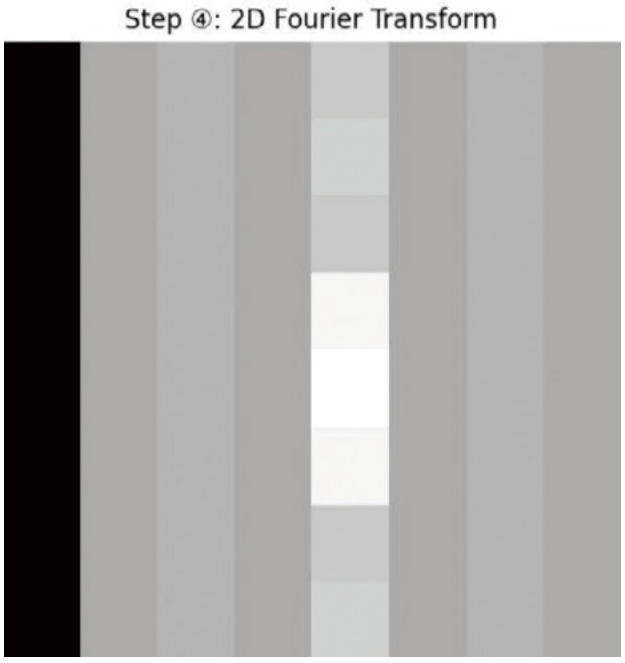


Fig.18　Step ④の写真

4.3　学習データのラベリング

分類モデルは通常、明確に区別できるカテゴリ（例えば、「未成熟」と「成熟」など）がある場合に効果的だが、成長の度合いは連続的なので、画像の特徴量の変化が微小であり、画像を明確なカテゴリに分類することは難しい。

2次元フーリエ変換処理した画像に対し、分類可能な特徴量が存在するか検証を行う。

分類（クラス分け）の定義

任意のネギ（写真撮影の都合上最前列に固定）を選び、ネギの高さを測る。ランダムで3つの束を選択肢、スケールで葉身の全長を計測し束の中のTOP 5の平均を出す。TOP 5の平均値の3束の平均を計算し50mm刻みで以下に分類とした。

クラス名：上記の方法による平均値のクラス分類

0：平均50mm未満
1：平均50mm以上100mm未満
2：平均100mm以上150mm未満
3：平均150mm以上200mm未満
4：平均200mm以上250mm未満
5：平均250mm以上300mm未満
6：平均300mm以上350mm未満
7：平均350mm以上400mm未満
8：平均400mm以上450mm未満
9：平均450mm以上500mm未満
10：平均500mm以上550mm未満

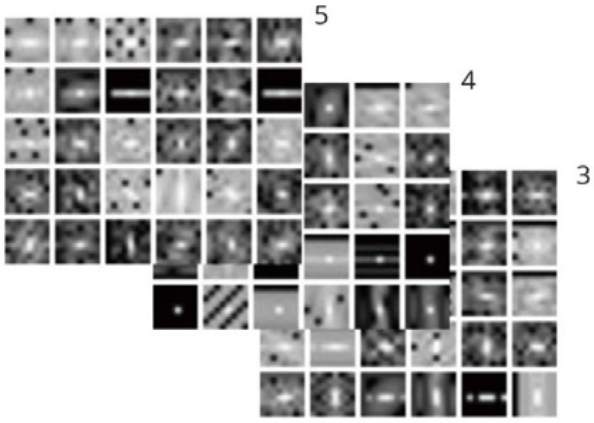


Fig.19　学習用のデータ

Fig.19のデータを用いて、モデルの構築と学習を行う。

下記にその時に用いたプログラムを列記する。

```
import tensorflow as tf
from tensorflow.keras import layers, models

# モデルの構築
model = models.Sequential()
model.add(layers.Conv2D(32, (3, 3),activation='relu', input_
shape=( 高さ , 幅 , チャネル数 )))
#・・・( 中間層を追加 )・・・
model.add(layers.Flatten())
model.add(layers.Dense(128, activation='relu'))
# 出力層：予測したい数値データの数だけニューロンを設
```

定
model.add(layers.Dense(4))# 例として 4 つの連続的な値を予測

# モデルのコンパイル
model.compile(optimizer='adam', loss='mse')

# モデルの学習
model.fit(トレーニングデータ , ラベルデータ , epochs= エポック数 , validation_data=(検証データ , 検証ラベル))

次に Google Colab 上での実行結果を Fig.20に示す。

```
5/15 - 1s - 55ms/step
- accuracy: 0.2835 - loss: 2.2767
Test accuracy: 0.2835164964199066
```

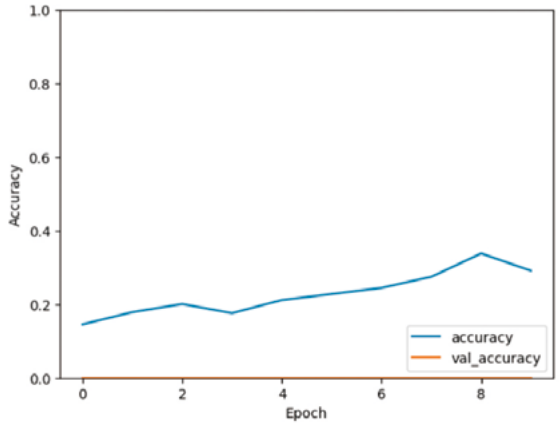


Fig.20　画像処理によるネギ高さの結果

以上の結果から

・学習が適切に進んでいない

バリデーションデータでの accuracy が 0 % のままであり、loss も改善していないことから、モデルがデータのパターンを適切に学習できていない可能性がある。

・過学習の懸念 :

トレーニングデータでの性能が若干向上している一方で、バリデーションデータでの性能が向上していないため、過学習が起きている可能性がある。

次に、 2 次元フーリエ変換画像ではなくカラー画像のモデルを用いて試みた。

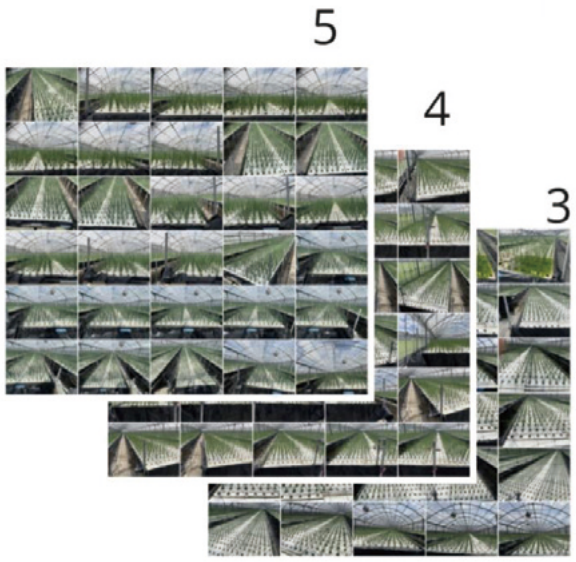
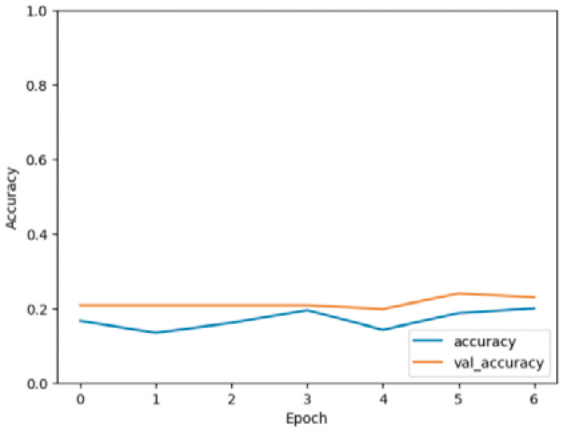


Fig.21　ラベリング

```
カラー画像に対応するためチャンネル数を3に変更（224x224x3）

モデルの構築（224x224x3に対応、Inputレイヤーを使用）
model = Sequential([
  Input(shape=(224, 224, 3)),  # Inputレイヤーを使用
  Conv2D(32, (3, 3), activation='relu'),
  MaxPooling2D(2, 2),
  Conv2D(64, (3, 3), activation='relu'),
  MaxPooling2D(2, 2),
  Flatten(),
  Dense(128, activation='relu'),
  Dropout(0.5),
  Dense(num_classes_new, activation='softmax')
])
```

Fig.22　カラー画像用に再設定するプログラム




```
3/3 - 0s - 25ms/step
- accuracy: 0.2083 - loss: 2.1943
Test accuracy: 0.2083333283662796
```

Fig.23　カラー画像用に再設定後の計算結果

結果は、過学習になっていると示唆される。
次に、パラメーターや拡張設定、既存モデルの転移学習

を設定して、再計算を行った。

```
# データ拡張の設定
datagen = ImageDataGenerator(
    rotation_range=30,
    width_shift_range=0.3,
    height_shift_range=0.3,
    zoom_range=0.3,
    horizontal_flip=True
)

# Pre-trainedモデル（MobileNetV2）を使う
base_model = MobileNetV2(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# 全ての層をフリーズ（事前学習済みの重みを保つ）
base_model.trainable = False

# モデルの構築（MobileNetV2 + カスタム層）
model = Sequential([
    base_model,
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(num_classes_new, activation='softmax')
])
```

Fig.24　パラメーターや拡張設定、既存モデルの転移学習を設定したプログラム

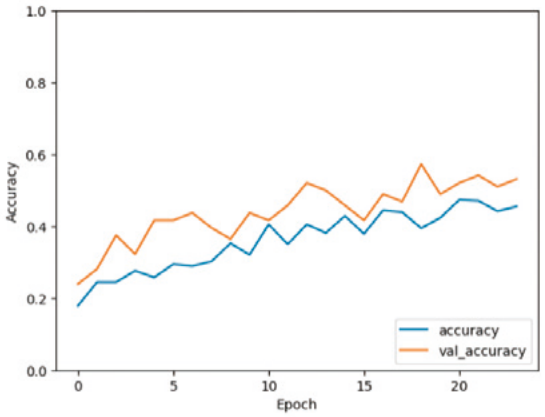


Fig.25　パラメーターや拡張設定、既存モデルの転移学習を設定したプログラムを用いた計算結果

以上の結果は、２次元フーリエ変換画像、カラー画像のモデルを用いたものに比べて大幅に改善した。

今後は、ネギの成長度合いが、連続的であり、やはり分類モデルは適切ではない可能性が高い。したがって、連続的な値として予測する回帰モデルの使用を検討する必要性がある。

5．地域の気象データの活用（OpenWeatherMapの利用）による天気のデータ収集

5. 1　ThingSpeak を用いたデータ処理

ThingSpeak のデータ処理は、Analytics, および、Actions で構成される。Analytics としては、MATLAB Analysis, MATLAB visualizations, Plugins があ る。Actions としては、ThingTweet, TimeControl, React, TalkBack, ThingHTTP があり、その概要を Fig.26に示す。

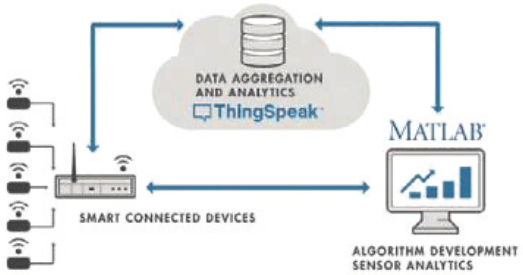


Fig.26　IoT 機器、ThingSpeak, Matlab/Simulink の概要[8)]

ThingSpeak 内に、チャネルを作り、チャネルセッティングを行う（Fig.27参照）。

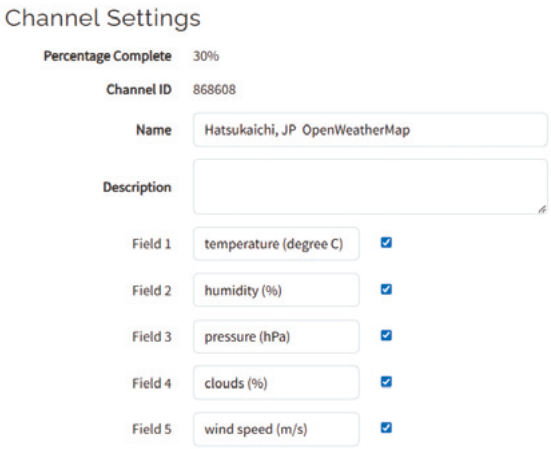


Fig.27　ThingSpeak のチャネル設定例

以上の設定を行い、OpenWeatherMap[9)] の環境データを Thingspeak に転送すると、広島工業大学近くの廿日市市の、気温（Fig.28)、湿度（Fig.29)、気圧（Fig.30)、雲の割合（Fig.31)、および風速（Fig.32）を示すことができる。

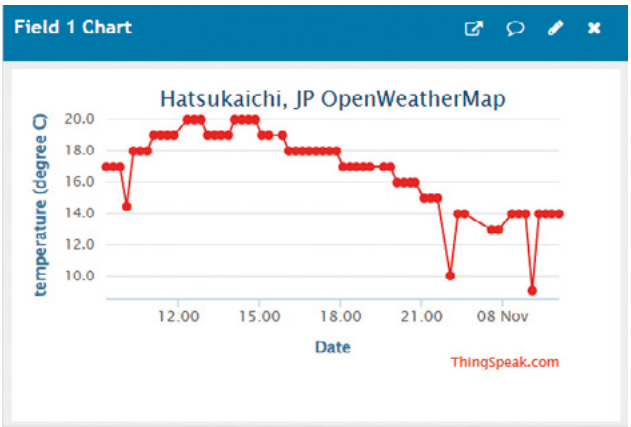


Fig.28　廿日市市の気温

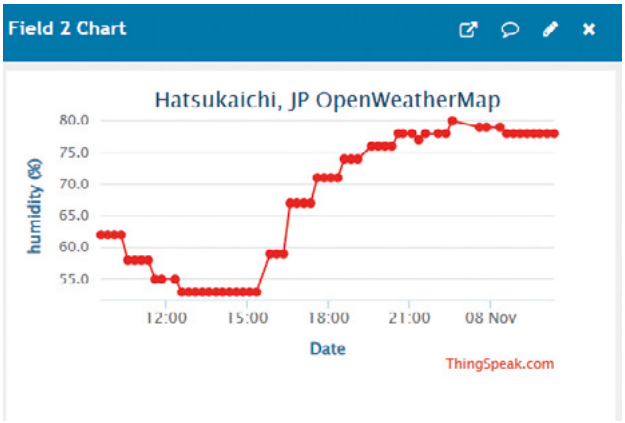


Fig.29　廿日市市の湿度

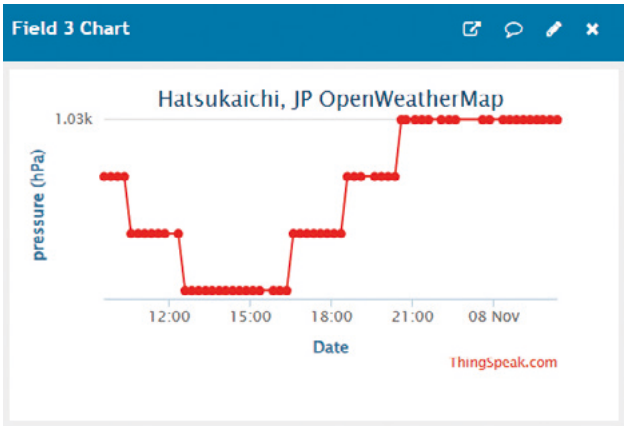


Fig.30　廿日市市の気圧

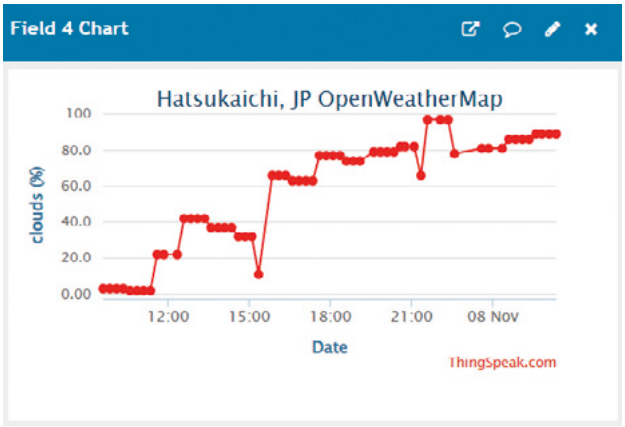


Fig.31　廿日市市の雲の割合

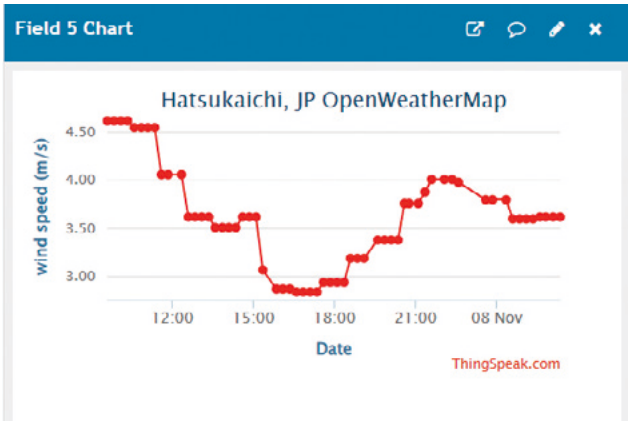


Fig.32　廿日市市の風速

以上より、OpenWeatherMap を用いることにより、地域の気象データを容易に収集でき、ネギの成長に与える環境の効果を考察できる可能性を示した。

まとめ

ネギの成長を観察するために、赤外線センサー、超音波センサなど各種センサーや、カメラで写真を撮影し、その画像を、２次元フーリエ変換、カラー画像のモデルの利用、および、パラメーターや拡張設定、既存モデルの転移学習を設定したプログラムを用いて計算を行った。その結果、パラメーターや拡張設定、既存モデルの転移学習を設定することにより、ネギの成長度合いの観察ができる可能性が示唆された。

また、OpenWeatherMap の環境データを Thingspeak に転送することにより、作物の成長に必要な環境データ（温度、湿度、気圧、雲の割合、風速など）が容易に入手し、ビジュアル化できることを示した。

謝辞

本研究を実施するにあたり、協力して頂いたネギ栽培業者（JA ひろしまクリーンカルチャーグループ）に謝意を表します。

文献

1）R. Matsuda, H. Suizu, K. Hiyama, K. Kai, I. Yamasaki, K. Oda, N. Okamitsu, K. Eto, T. Tanaka, Industry 4.0, 3（2018）, Issue 1, pg（s）11-14

2）T. Tanaka, N. Okamitsu, T. Hamada, K. Nishino, Electrotechnica & Electronica55（E+E）,（3-4）, 2020, 35-40, ISSN: 0861-4717（Print）, 2603-5421（Online）

3）T. Tanaka, N. Okamitsu, K. Nishino, Electrotechnica & Electronica（E+E）, 55（5-8）, 2020, 94-98, ISSN: 0861-4717（Print）, 2603-5421（Online）

4）N. Okamitsu, K. Miyazono, T. Tanaka, Electrotechnica & Electronica (E+E), 55 (9-12), 2020, 99-104, ISSN: 0861-4717 (Print), 2603-5421 (Online)
5）T. Tanaka, M. Yamauchi, K. Kakugawa, T. Uchida, S. Ishihara, K. Ishihara, Electrotechnica & Electronica (E+E), 57 (1-2), 2022, 24-30, ISSN: 0861-4717 (Print), 2603-5421 (Online)
6）https://www.osepp.com/electronicmodules/sensor-modules/62-osepp-ultrasonicsensor-module
7）https://www.marutsu.co.jp/pc/i/2189847/?srsltid=AfmBOopqOkq0sAitr7YWv44En6JUO8sKIjyR97Yek2FP7_zEqa1F7tns
8）https://jp.mathworks.com/products/thingspeak.html
9）https://openweathermap.org/