ユーザーズマニュアル

物体認識システム

発行日 2022 年 3 月 31 日

公立大学法人会津大学
株式会社東日本計算センター

目次

1. はじめに

1.1. 物体認識システムとは

ロボットナビゲーションにおいて、事前に経路を計画しロボットを自動走行させますが、走行経路における想定外の障害がある場合に経路を再計画し走行する必要があります。その経路を計画する際にはロボットの位置情報や障害物の位置情報も必要になるため、その情報を外部環境のカメラから計測しロボットへ提供させる必要があります。

2021 年度は LICTiA 1F に設置されたマルチ監視カメラシステムから取得したカメラキャプチャーを基に物体認識を行い、特定の基準から認識対象(ロボットや人、机、椅子)の位置を計測します。また、算出した推定位置と物体情報をもとにレイアウト位置の監視をおこないます。認識した物体情報および推定位置情報を RDR (Robot Data Repository)へ送信し、データの蓄積を実施します。

1.2. 本書の記載範囲

本書の記載範囲は、物体認識システムを使用するユーザー向けの操作マニュアルになります。

本システムの環境構築に関しては「インストールマニュアル_物体認識システム」を参照してください。

1.3. 動作環境

本システムの動作環境を記載します。
依存ライブラリバージョンは本システム検証時点となります。

表 1-1 使用機器一覧

環境		バージョン	補足
OS	Ubuntu Server	18.04 LTS	-
CPU	Intel Core i7 8650U 4.2GHz	-	4 コア以上推奨
GPU	8GB[1]	-	
メモリ	64GB 以上	-	-
ストレージ(SSD)	512GB 以上	-	-
開発言語	Python	3.7 系	-
依存ライブラリ	tensorflow-gpu	1.15.0	GPU 版
	Keras	2.3.1	-
	YOLO	V4	-
	Cython	0.29.23	-
	gast	0.4.0	-
	paddlepaddle	1.8.4	-
	pycocotools	2.0.2	-
	pydot	1.4.2	-
	numpy	1.19.5	-
	tensorboard	1.15.0	-
	torch	1.8.1	-
	opencv-python-haedless	3.4.9.33	-
	testresources	2.0.1	-
	h5py	2.10.0	-
	urllib3	1.22	-

[1] GeForce RTX2080 SUPER 8GB×2

1.4. 前提事項/注意事項

本システム導入にあたっての前提ならびに注意事項を下表に示します。

表 1-2 前提ならびに注意事項

前提事項	(1)　インストールマニュアル(別紙)に沿って、システム構築が完了していること (2)　MQTT Broker が起動していること (3)　マルチ監視システムが起動していること (4)　RDR が起動していること
注意事項	無し

1.5. 使用機器

本システムで使用する機材を次の表に記載します。

表 1-3 使用機器一覧

No.	使用機器	個数	補足
1	パーソナルコンピューター	1	-

1.6. 関連資料

本システムと関連する資料を次の表に記載します。

表 1-4 関連資料一覧

No.	資料名
1	インストールマニュアル_物体認識システム

2. 本システムでできること

本システムはマルチ監視カメラシステムから取得したカメラキャプチャーに対して物体認識を行い、物体の認識と推定位置を算出します。また、算出した推定位置と物体情報をもとにレイアウト位置の監視をおこないます。認識した物体情報および推定位置情報を RDR に蓄積します。

3. 動作手順

3.1. カメラキャリブレーションファイルの設定

マルチ監視カメラシステム上のカメラリスト情報とカメラキャリブレーション情報を設定してください。以下にカメラリスト情報を設定する camera_conf.json とカメラキャリブレーション情報を設定する cameraX_calibration.json の設定項目を記載します。

表 3-1 camera_conf.json ファイル構成

スペック	フォーマット
ファイル形式	JSON
項目	**説明**
CameraId	マルチ監視カメラシステムに接続されている IP カメラの MAC アドレスを設定してください。
Calibration	カメラキャリブレーションファイル名を設定してください。

```
[
    {"CameraId": "xx-xx-xx-xx-xx-xx", "Calibration": "Camera1_Calibration.json"},
    {"CameraId": "xx-xx-xx-xx-xx-xx", "Calibration": "Camera2_Calibration.json"},
    {"CameraId": "xx-xx-xx-xx-xx-xx", "Calibration": "Camera3_Calibration.json"},
    {"CameraId": "xx-xx-xx-xx-xx-xx", "Calibration": "Camera4_Calibration.json"}
]
```

図 3-1. camera_conf.json の記載内容

表 3-2 cameraX_calibration.json ファイル構成

スペック		フォーマット
ファイル形式		JSON
項目		**説明**
internal_params		内部パラメータを設定する項目になります。
	fx	カメラ行列を設定してください。
	fy	
	cx	
	cy	
	k1	カメラの歪み係数を設定してください。
	k2	
	p1	
	p2	
	k3	
external_params		外部パラメータを設定する項目になります。
	rx	回転行列を設定してください。
	ry	
	rz	
	tx	並進ベクトルを設定してください。
	ty	
	tz	
camera_id		カメラを識別する ID を設定してください。

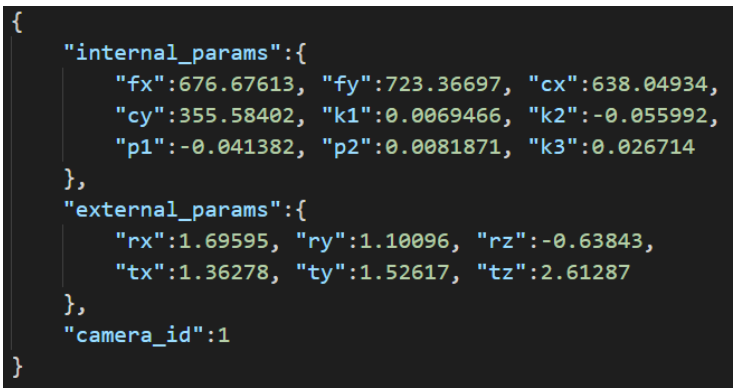

図 3-2. camera_conf.json の記載内容

3.2. 接続設定

物体認識システムがアクセスするマルチ監視カメラシステム、RDR、MQTT Server などの接続先情報を設定する必要があります。

下図の赤枠で囲んである箇所に実行環境に合った設定値を記載してください。

- object_recognition_controller.py

表 3-3 object_recognition_controller.py の設定項目

項目	説明
self._mqtt_host	MQTT Service の接続ホスト名
self._mqtt_port	MQTT Service の接続ポート名
self._upload_api_url	RDR の画像アップロード WEB API

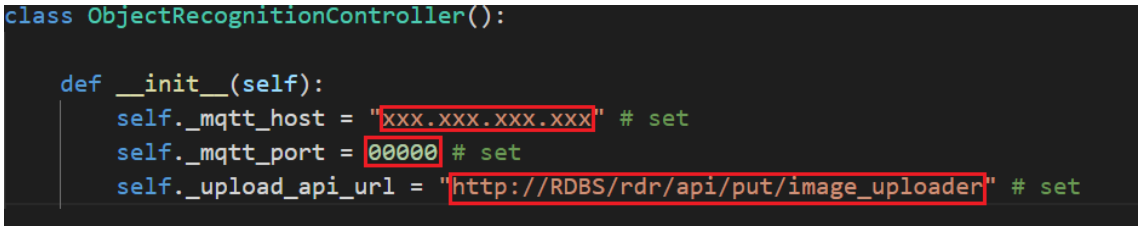

図 3-3. object_recognition_controller.py の設定箇所

- web_cam_interface.py

表 3-4 web_cam_interface.py の設定項目

項目	説明
self._user	マルチ監視カメラシステムユーザーを設定
self._password	マルチ監視カメラシステムパスワードを設定
self._host	マルチ監視カメラシステムホスト名を設定
self._port	マルチ監視カメラシステムポート名を設定

```
class WebcamInterface():

    def __init__(self, camera_id_a, identification_id_a):
        self._user = 'user'             # set
        self._password = 'password'     # set
        self._host = 'xxx.xxx.xxx.xxx' # set
        self._port = 'xxxx'             # set
        self._camera_id_a = camera_id_a
        self._identification_id_a = identification_id_a
        self._http_pool = urllib3.PoolManager()
```

図 3-4. web_cam_interface.py の設定箇所

- layout_monitoring.py

表 3-5 layout_monitoring.py の設定項目

項目	説明
service_host	MQTT Service の接続ホスト名
service_port	MQTT Service の接続ポート名
self._request_api_url	RDR のレイアウト情報取得 WEB API

```
class LayoutMonitoring():

    def __init__(self):
        service_host = "xxx.xxx.xxx.xxx" # set
        service_port = 00000 # set
        self._request_api_url = "http://RDBS/rdr/api/get/layoutInfo" # set
```

図 3-5. layout_monitoring.py の設定箇所

3.3. アプリケーションの起動

ターミナルを起動し、以下のコマンドを実行することでアプリケーションを起動します。

```
$ python3 object_recognition.py
```

3.4. アプリケーションの停止

アプリケーション実行中にターミナルから Ctrl+C を押下するとアプリケーションが停止します。

```
$ Ctrl+C
```

4. エラーメッセージ

物体認識システム実行中にエラーが発生した場合は以下のメッセージがコンソール上に出力されます。

表 4-1 エラーメッセージ一覧

No.	項目	説明
1	Unable to get camera capture	マルチ監視カメラシステムにアクセスが失敗した場合に出力されるエラーです。
2	Unable to upload camera capture	RDR にカメラキャプチャーの送信が失敗した場合に出力されるエラーです。
3	Unable to update estimated position data for object	RDR に推定位置情報の送信が失敗した場合に出力されるエラーです。
4	Unable to get layout data	RDR からレイアウト情報の取得が失敗した場合に出力されるエラーです。
5	Unable to update layout data	RDR にレイアウト情報の送信が失敗した場合に出力されるエラーです。

5. FAQ

次によくある質問を一覧で記載します。

表 5-1 FAQ 一覧

No.	Q 質問	A 回答
1	-	-

著作権