研究論文

MZ Platform を用いたロボット自動化システムの IoT 化

木村宏樹[*1]、酒井昌夫[*1]、島津達哉[*1]、平出貴大[*1]

IoT Application of Robotic Automation System Using MZ Platform

Hiroki KIMURA[*1], Masao SAKAI[*1], Tatsuya SIMADZU[*1] and Takahiro HIRADE[*1]

Industrial Research Center[*1]

中小企業支援におけるロボットや IoT 活用の例示のため、安価で手軽に利用可能な既存技術を活用して、組立・検査のロボット自動化システムの構築とその IoT 化を行った。Raspberry Pi やオープンソースソフトウェア等を用いて、ロボットや周辺機器を制御する PLC と組立良否を AI 画像判定するシステムを構築し、(国研)産業技術総合研究所の IoT ツール「MZ Platform・スマート製造ツールキット」を用いて、検査不良数のカウントとその遠隔モニタリングを実現する IoT 化を行った。

1. はじめに

製造現場における生産性向上を目的にロボットや IoT 活用の取組が進められる中、中小企業においてもこれらの技術を容易に利用できることが求められる。ロボットシステムや IoT システムの構築は、SIer(システムインテグレータ)などのベンダー企業が担うのが一般的だが、低価格で高機能なハードウェアの普及やソフトウェア開発手法の発展により、小規模なシステムの自作も難しいことではなくなってきている。

ロボットに関してはオープンソースソフトウェアやミドルウェア等、IoT に関しては各種支援ツールやモジュール化されたセンサ等によるシステム構築の簡易化や効率化の流れがある。ただし、中小企業に対して、これらの技術にアクセスする機会や習得へのモチベーションを高める機会が十分に提供されているとは言えない。

このため、これらの技術の活用の具体例を利用可能な既存技術を用いて具現化し、例示することは、本分野の企業支援に取り組むうえで重要であると考えられる。そこで本研究では、安価で手軽に利用可能な既存技術を活用して、アームロボットを用いた組立・検査の自動化システムの構築とその IoT 化を行った。具体的には、シングルボードコンピュータ「Raspberry Pi」やオープンソースソフトウェア等を用いて、ロボットや周辺機器を制御する PLC (Programmable Logic Controller)と組立良否を AI 画像判定するシステムを構築した。このロボット自動化システムを対象に、(国研)産業技術総合研究所の IoT ツール「MZ Platform・スマート製造ツールキット」[1)~3)]を用いて、検査不良数のカウントとその遠隔モニタリングを実現する IoT 化を行った。

2. 実験方法

2.1 ロボット自動化システムの構築

2.1.1 ロボットによるボールペンの組立と検査の自動化

図 1 に構築した組立・検査のロボット自動化システムを示す。本システムでは、アームロボットがボールペンを組み立て、キャップのねじ込みの状況を撮影したカメラ画像に対して AI 推論により良・不良を判定し、不良の場合に表示灯が点灯する。システムは、大きく「組立」と「検査」の工程に分けられる。それぞれの技術的な要素である PLC、AI について、システムを構築した。

2.1.2 PLC による制御

ロボットやその周辺機器は、一般に産業用 PLC でシステム構築される。産業用 PLC は高価であり、機器を制御するためのプログラム言語も機種により異なるため、未経験の企業等が使用するには簡単なものではない。本研究では、Raspberry Pi(3B+)と DC24V アイソレート I/O 基板を使用し、ロボットやコンベア、表示灯、光電センサ等を制御する PLC を構築した(**図 2**)[4)]。PLC の制御には、ラダー・プログラムの開発環境であるオープンソースソフトウェア「OpenPLC」を使用した [4)]。作成したプログラムの一例を**図 3** に示す。センサの入力信号に対して、アームロボットの駆動や表示灯の点灯などを制御することができる。アームロボットには、人協働ロボット COBOTTA ((株)デンソーウェーブ製)を使用した。3D プリンタで作製した専用のハンドと治具により、ボールペンのキャップの把持・組立を行う動作を教示した。

2.1.3 AI による画像分類

組立後のボールペンの良・不良を AI によって判定するシステムを構築した。AI 学習には、Windows 環境で

*1 産業技術センター　自動車・機械技術室

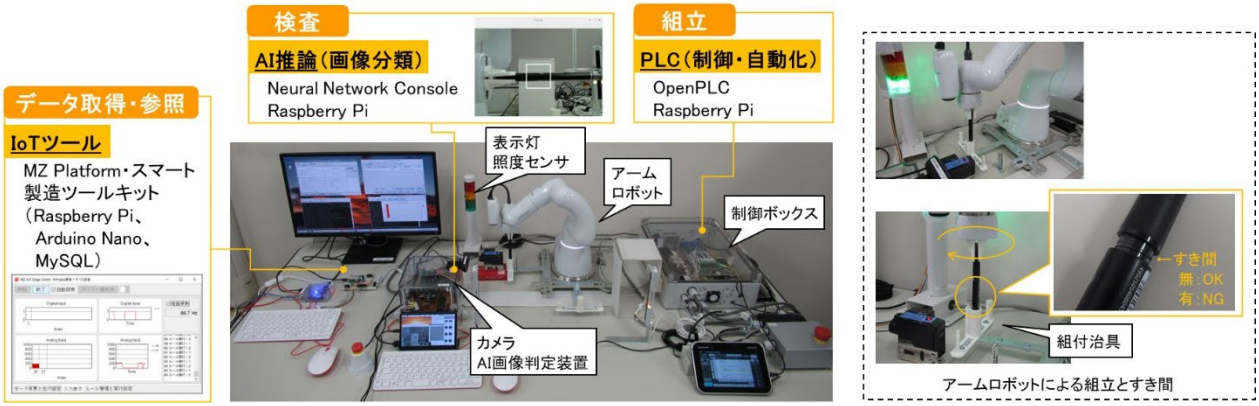

図 1　組立・検査のロボット自動化システム

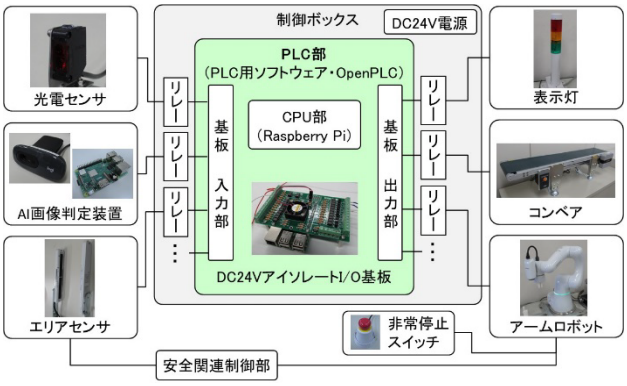

図 2　構築した PLC

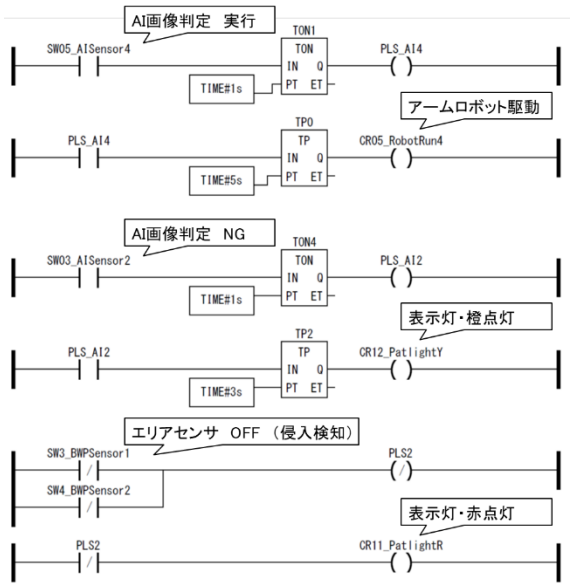

図 3　OpenPLC で作成したラダー・プログラム

動作する「Neural Network Console」（ソニーネットワークコミュニケーションズ(株))[5),6)]を使用した。本アプリは、ドラッグ&ドロップで簡単にニューラルネットワークを設計でき、ディープラーニングを用いた高度な AI 開発が可能である。無料で使用でき(クラウド版は有料)、また、学習結果を Python コードで出力できるため、マイコンボード等への実装も可能である。

本研究では、畳み込みニューラルネットワーク(CNN:Convolutional Neural Network)を設計し、ボールペンのすき間画像からすき間の良否を学習させた(**図 4**、**図 5**)。学習に用いたデータセット(画像サイズ 40 px ×40px、学習用 38 枚、評価用 22 枚)は小規模であり、ごく短時間で処理できた。学習結果を Python コードで出力し、PLC とは別の Raspberry Pi(3B+)に実装した。AI 推論結果が不良と判定された場合、Raspberry Pi の I/O からの出力信号が PLC に入力され、表示灯が点灯するよう PLC のラダー・プログラムを作成した。

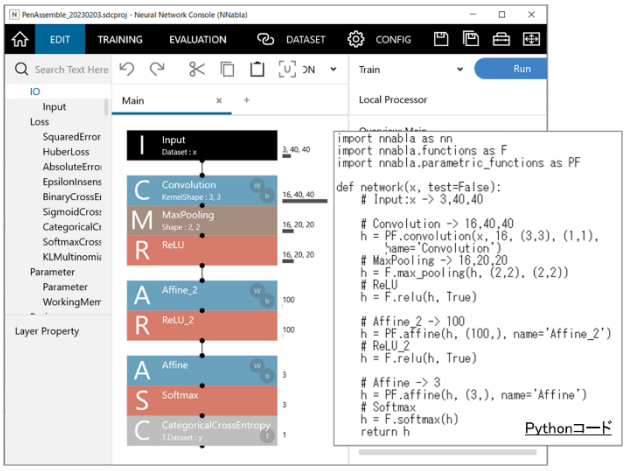

図 4　Neural Network Console と設計した CNN

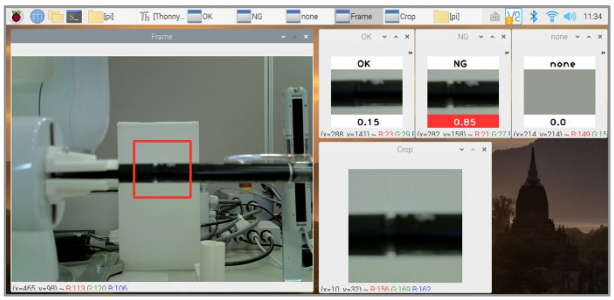

図 5　AI 推論の実装

2.2 MZ Platform による IoT 化

2.2.1 MZ Platform・スマート製造ツールキット

MZ Platform は、中小企業の IT 化支援を目的に開発されたソフトウェア開発ツールである。高度なスキルを必要とせずに、コンポーネントと呼ばれるソフトウェアの部品を画面上に組み合わせることで、ソフトウェアを作成することができる。これを拡張した「スマート製造ツールキット」は、ソフトウェアだけでなくハードウェアまで含めた IoT システムの構築を可能とする IoT ツールである。具体的には、安価なセンサやマイコンを使用した計測・可視化などのシステムを自作することができ、そのためのアプリケーション(MZApp)も提供されている。これらのツールは、MZ プラットフォームユーザー会 [2]から配布され、会員登録すれば無償で利用することができる。操作マニュアルや開発チュートリアルも充実しており、独自アプリの開発も可能である。

2.2.2 IoT システムの構築

スマート製造ツールキットを使用して構築した IoT システムの構成を**図 6** に示す。機器の稼働状況をデータ計測部(エッジ PC)で取得し、データ蓄積部(サーバ PC)に送信する。モニタリング部(クライアント PC)からサーバ PC のデータベースにアクセスすることで、LAN 内において稼働状況の遠隔モニタリングが可能である。

具体的には、Raspberry Pi(4B)をエッジ PC とし、アナログ・デジタル入出力可能なマイコン「Arduino Nano」を接続した。スマート製造ツールキットの計測用アプリ「IoTEdgeApp」を利用して、接続したセンサの出力値を取得する。本アプリでは、取得したセンサ値に対する閾値と識別番号を設定する。センサ値が設定した閾値ルールを満たした場合、識別番号がサーバ PC のデータベースに登録される。データベースは、スマート製造ツールキットが提供するもので構築した。クライアント PC からデータベースにスマート製造ツールキットの可視化用アプリ「IoTDBViewer」を利用してアクセスする。本アプリでは、データベースに登録された識別番号の登録回数をカウントする。カウント数を時間・日・月別でグラフ表示することも可能である。

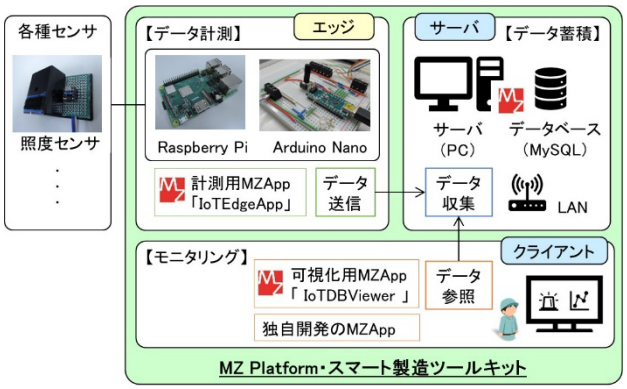

図 6　構築した IoT システム

2.2.3 ロボット自動化システムへの IoT システムの実装

スマート製造ツールキットを用いて、今回構築したロボット自動化システム(図 1)に IoT システムを実装した(**図 7**)。ロボット自動化システムでは、キャップのすき間が不良(NG)の場合、表示灯が点灯(橙色)するため、これを照度センサで検知する。表示灯に照度センサを設置し、エッジ PC の計測用アプリ「IoTEdgeApp」でセンサ出力値に対する閾値ルールと識別番号を設定した(図 7(a))。この閾値ルールに基づき、表示灯の点灯の有無を判定し、識別番号がデータベースに登録される。クライアント PC の可視化用アプリ「IoTDBViewer」を用いてデータベースにアクセスし、識別番号の登録回数を参照することで、不良数をカウントする。

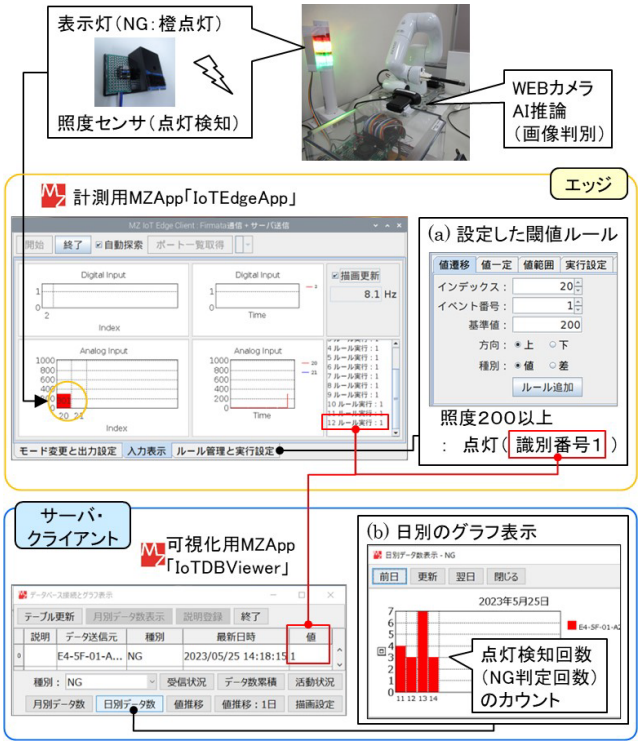

図 7　IoT システムの実装

2.2.4 センサ出力値の保存

計測用アプリ「IoTEdgeApp」では、センサ出力値はアプリの画面上での表示と閾値ルールの判定に利用されるが、その値は保存されない仕様となっている。「データ活用」を考えた場合、センサ値を保存できることが求められる。そこで、アプリを改良し、センサ値を一定周期でエッジ PC に保存する機能を追加した(**図 8**)。本アプリでは、設定したサンプリング周期で、設定したデータ数分のセンサ値を保持し、一度にまとめて保存する。

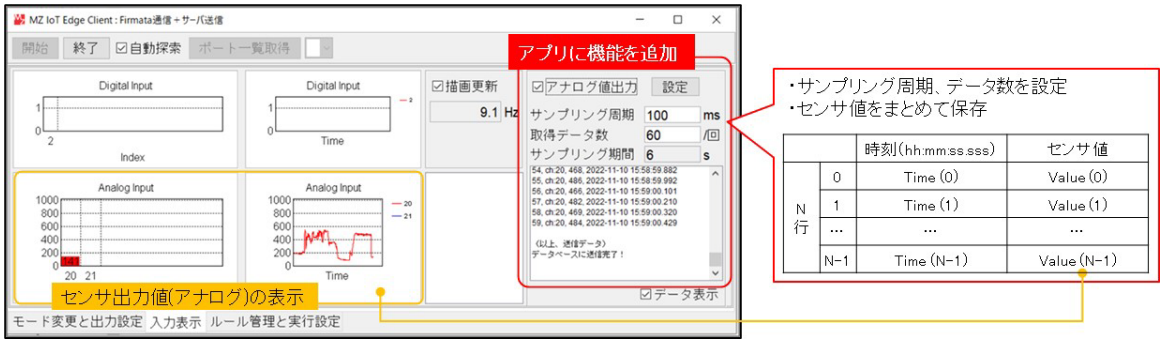

図 8　計測用アプリの改良

3. 実験結果及び考察

3.1 システムの動作確認

構築したシステムの動作確認のため、ボールペンの組立と検査、不良数の遠隔モニタリングを行った。クライアントPCの可視化用アプリ「IoTDBViewer」で表示した日別のグラフから、不良数をカウントできることを確認した(図 7(b))。

3.2 センサ出力のサンプリング周期の評価

センサ出力値の保存について、設定したサンプリング周期に対するセンサ値の取得周期を評価した。Raspberry Pi(3B+)をエッジ PC とし、センサ値(データ数 60)を時刻情報とともに保存し、サンプリング周期を評価した(**表**)。設定周期に対する実際の周期の結果から、サンプリング周期 80ms を下限としてセンサ値を保存できることが分かった。本アプリによる具体的な測定例として、アームロボットで加速度センサを把持して旋回(約 1 回/秒)させた際のセンサ値(サンプリング周期 100ms、データ数 50、5 秒間)を**図 9** に示す。概ね一定周期でセンサ値を保存できることが確認できた。

表　サンプリング周期の評価結果

設定周期 (ms)	実際の周期(ms)	
	平均値	標準偏差
500	500	1
200	199	9
100	100	2
80	81	9
50	66	12
25	68	16

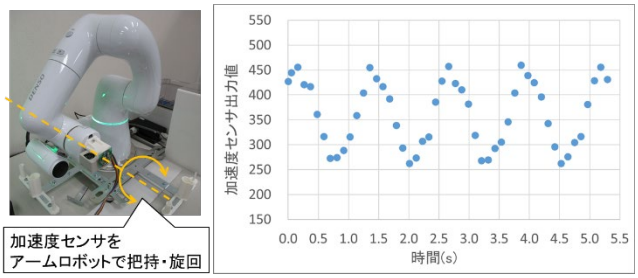

図 9　センサ値の取得例

4. 結び

本研究の結果は、以下のとおりである。

(1) 安価で手軽に利用可能な既存技術を活用して、組立・検査のロボット自動化システムを構築した。

(2) ロボットや周辺機器を制御する PLC と AI 画像判定システムを Raspberry Pi やオープンソースソフトウェア等で構築した。

(3) 「MZ Platform・スマート製造ツールキット」を用いて検査不良数のカウントとその遠隔モニタリングをした。また、センサ出力値を一定周期で保存する機能を追加し、その取得周期を評価した。

文献

1) 澤田浩之, 徳永仁史, 古川慈之: Synthesiology, **8**(3), 158(2015)

2) 国立研究開発法人　産業技術総合研究所　製造技術研究部門: MZ プラットフォームユーザー会, https://ssl.monozukuri.org/mzplatform/. (2023/05/15)

3) 古川慈之: 人工知能学会第二種研究会資料, **2017**(KST-32), 1(2017)

4) 今関雅敬: ラズパイで PLC ハード&開発環境編, 21(2021), CQ 出版社

5) 足立悠: ソニー開発の Neural Network Console 入門, 61(2018), 株式会社リックテレコム

6) ソニーネットワークコミュニケーションズ株式会社: Neural Network Console, https://dl.sony.com/ja/. (2023/05/15)