

人とロボットの協働による工場のスマート化支援

ー協働ロボットによるバリ取り作業自動化に関する研究ー

Support for the realization of smart factories by the collaboration between humans and robots

- Research on automation technology in deburring work by a collaborative robot -

技術開発部 プロジェクト研究科 安藤久人 松本聖可

技術開発部 生産・加工科 鈴木健司 菅野雄大 柿崎正貴

ものづくりの現場では、生産年齢人口の減少により、AI・IoT、ロボット等を導入したスマート工場の実現が進められている。課題は、中小企業がこれら技術を導入して十分な効果が得られるかを事前に検証することが難しい点にある。本研究ではバリ取り作業に注目し、協働ロボットを用いた省力化の検証と作業の可視化に取り組んだ。シミュレーションによる画像処理とロボットの動作生成、バリ取り作業時の負荷力を実測した。

Key words: スマート工場、省力化、協働ロボット、バリ取り

1. 緒言

少子高齢化により、我が国の生産年齢人口（15歳～64歳）は1995年をピークに減少している。国勢調査によると、2015年の生産年齢人口は7,629万人であるが、推計によれば10年後の2030年には6,875万人（9.88%減）、2060年には4,793万人（37.2%減）にまで減少するとされている¹⁾。本県においても、10年後の生産年齢人口は推計で106万人から95万人と10.4%減少すると予想されている²⁾。このような社会背景の中で、ものづくりの現場では、AI（人工知能）やIoT（あらゆるものをインターネットで繋ぐこと）、ロボット等を導入した工場のスマート化が進められている。

しかし、課題として、豊富な資金力や人材を有する大企業とは異なり、中小企業がこれらの先端技術を導入し効果が得られるかどうかの事前検証が難しい点が挙げられる。そこで我々は、主に中小企業がロボット導入を事前に検証することができるテストベッドを構築した。本研究の目的は、企業の個別課題であるバリ取り作業の省力化である。今年度は、シミュレーションによる動作生成と作業負荷の可視化に取り組んだ。

2. 研究

2. 1. テストベッドの概要

ロボット導入の個別課題を評価するためにロボットのテストベッドを構築した。既存の産業用ロボットである6軸垂直多関節ロボット（RV-20F-D、三菱電機株式会社）の他に、新しく協働ロボット（UR10e、ユニバーサルロボット社）を導入した。産業用ロボットは、一般的に高速かつ高剛性であるが安全上、柵が必要となる。一方、近年は、人手不足から人が行う作業の一部をロボットに置き換え、人との協働作業をロボット

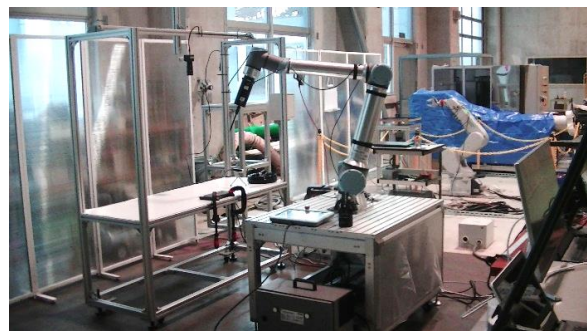
にさせたいという要望が多い。それには安全柵は邪魔になる。協働ロボットは接触センサ等が内蔵されており人との接触を検知し停止するなどの安全対策が講じられている。基本的にはリスクアセスメント上問題が無ければ安全柵を設置しなくてよい。そこで我々は、人とロボットによる協働作業等を検証するため当該ロボットを導入した。図1にロボット、及び構築したロボットテストベッドを、表1には各ロボットの仕様を示す。



(a) 産業用ロボット



(b) 協働ロボット



(c) 全体図

図1 ロボットテストベッド

表1 ロボットの仕様

	産業用ロボット	協働ロボット
型式	RV-20F-D	UR10e
自由度	6	6
可搬重量	20 kg	10 kg
最大速度 ^(*)	110~360 度/s	120~180 度/s
最大リーチ半径	1,094 mm	1,300 mm
本体質量	120 kg	33.5 kg
安全柵	要	不要

(*) 各軸で最大速度は異なります。

2. 2. バリ取りの省力化

2. 2. 1. バリ取り作業省力化の課題

本研究では、具体的な企業のニーズをもとに、協働ロボットによるバリ取り作業の省力化について検証を行った。

バリ取り作業は、ワークが重い、バリ取り部の形状が複雑、バリ取り部の数が多い、ワークの向きを変えたり反転させる必要がある、単調な作業が長時間続くなどのため省力化ニーズや課題が多い。

2. 2. 2. 課題への対応

本研究では第1のステップとして、平板にあけた穴のバリ取り作業を対象を絞った。

バリ取り作業のシナリオとして、(1) 鉄板に穴が機械であけられる、(2) そのワークを人がバリ取り用の作業台に置く、(3) 置かれたワークの穴をロボットがカメラで認識する、(4) その穴にロボットアーム先端が移動しバリを除去する、(5) バリが除去されたら人がワークを裏返す、(6) 裏返されたワークの穴をロボットがカメラで認識する、(7) その穴にロボットアーム先端が移動しバリを除去するという一連の工程とした。

本研究では、まず初めに上述のシナリオの個別のタスクを実行するために必要な要素技術についてコンピュータシミュレーションにより検証した。具体的には、シナリオ(3) 実現のために穴の位置を画像処理で認識できるか、シナリオ(4) 実現のために認識した穴の位置にロボットのアーム先端を移動させる軌道を生成できるか、シナリオ(6)、(7) 実現のために穴の位置がオフセットされた場合を認識し、オフセットされた位置にアーム先端が追従できるかの3つである。

3. 実験

3. 1. シミュレーション

3. 1. 1. 穴形状の認識とロボット動作生成

穴形状の認識には、写真で撮影した穴の画像を用い

た。表2に実験で用いた画像の条件を示す。また、図2に画像上の座標を示す。

穴形状の認識には、画像処理で広く用いられているオープンソースのコンピュータ・ビジョン・ライブラリOpenCVを活用した。また、コードはPythonにより記述した。

ロボットの動作生成には、ユニバーサルロボット社のロボットモデルと、広く普及しているロボット開発のミドルウェアであるROSを用いた。ロボットはアームロボットの軌道計算ライブラリであるMoveIt!を用いて動かし、ROS標準の可視化ツールであるrvizで表示した。コードはC++で記述した。

表3にこのときの開発環境を示す。

3. 1. 2. 穴位置オフセットへの追従

画像処理を行う穴の位置がオフセットされた場合に、オフセットを認識し、その変位を自動で追従できるか検証した。使用した画像は表2に示す画像の穴の位置を中心からX方向に80mm、Y方向に45mmオフセットさせたものである。

表2 画像の条件

形状	個数	位置	直径[mm]	画素数[pixel]
円	1	中央	80 mm	960×540

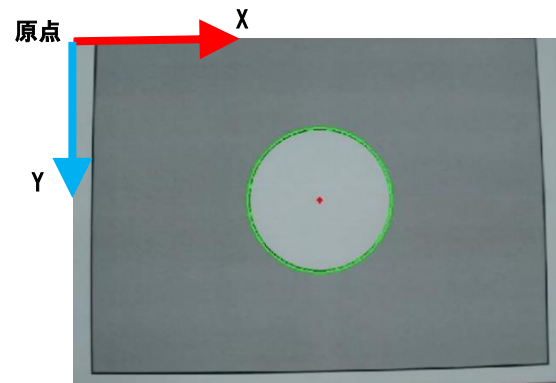


図2 画像上の座標系

表3 開発環境

OS	ROS	画像処理	コード
Ubuntu16.04 64bit	Kinetic Kame	OpenCV 3.1.0	Python C++

3. 2. バリ取り作業負荷の測定

作業の省力化のためには、その作業負荷の可視化が必要不可欠である。そこで、図1、表1に示す協働ロボット(UR10e、ユニバーサルロボット社)によりバリ取り作業を行い、その際に工具先端に加わる作業負荷を測定した。使用した工具は、ブレード式バリ取りア

クチュエータ (CDB-8-11-0、SCHUNK 社) を用いた。

図3に使用したワークを示す。鉄板(横200×縦100×厚さ10[mm])に直径13[mm]のドリルで穴をあけた。工具移動速度は約6秒/回転とした。

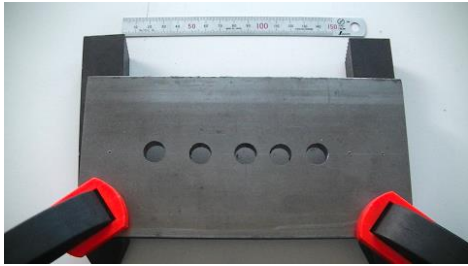


図3 バリ取り実験用ワーク

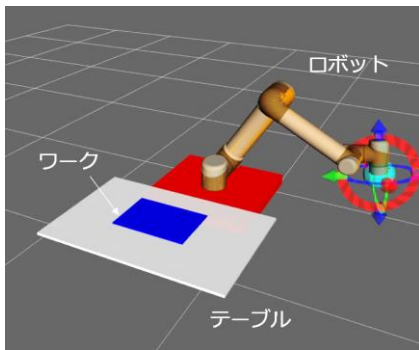
4. 結果

4. 1. シミュレーション

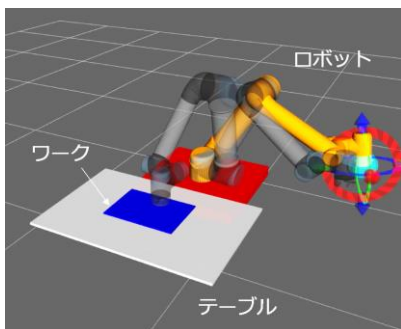
4. 1. 1. 穴形状の認識とロボット動作生成

円の中心座標 (x, y) をそれぞれ、 $x = "x_circle"$ 、 $y = "y_circle"$ という独自型変数として定義した。これらを /Circle という ROS 上で扱うことができるトピック変数として格納した。また、移動させたい点を $x = "x_param"$ 、 $y = "y_param"$ と定義しロボット側に読み込ませた。これら一連の動作を ROS で実行した。図4にこのときの動作を示す。

仮想的に青いワーク上に穴の画像が配置されているとする。穴の画像を認識することで、青で示すワーク中央にロボットアームの先端が移動した。



(a) 画像認識前

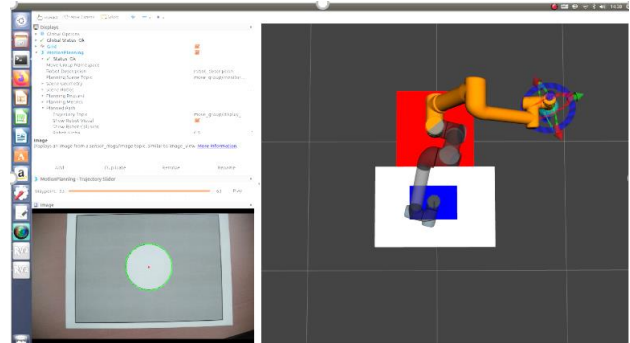


(b) 画像認識後

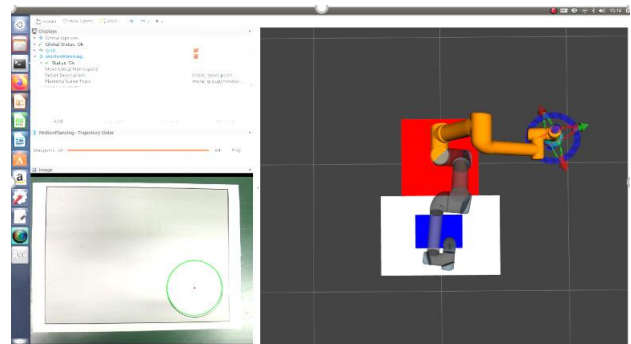
図4 穴位置の認識とロボットの動作軌跡

4. 1. 2. 穴位置オフセットへの追従

穴の位置をオフセットさせた場合のロボットアーム先端の動きを図5に示す。図5.(a)が穴の位置が画像の中央の場合、図5.(b)が穴の位置がオフセットされた場合である。アーム先端の位置が読み込む画像の穴の位置に追従していることが分かる。



(a) 穴の位置が画像中央の場合



(b) 穴の位置がオフセットされた場合

図5 ロボットの動作軌跡

4. 2. バリ取り作業負荷の測定

協働ロボットとワークの配置、及び座標系を図6に示す。ブレード式のバリ取りアクチュエータにより、ワーク中央の直径13[mm]の穴のバリ取りを行った。

バリ取り作業時のアーム先端に加わる負荷は、ロボットの状態をモニターするソフトウェア (UR Log Viewer version 1.2.1.0) を用いて取得した。この時サンプリング周波数を100[Hz]とした。

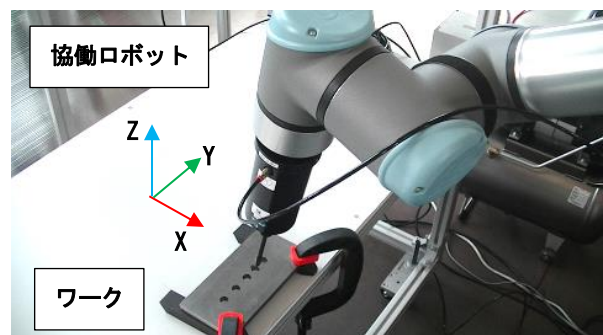


図6 協働ロボットとワークの座標系

図7に、バリ取り作業時に測定したアーム先端に加わる力の時間変化を示す。X軸、Y軸、Z軸方向に作用する負荷力をそれぞれ F_x 、 F_y 、 F_z とした。また、矢印の方向が負荷力の正の方向である。

図7より、最大でX軸方向に 21.4[N]、Y軸方向に 12.1[N]の力が作用し、 F_x 、 F_y の値が時間と共に周期的に変動していることが確認できた。

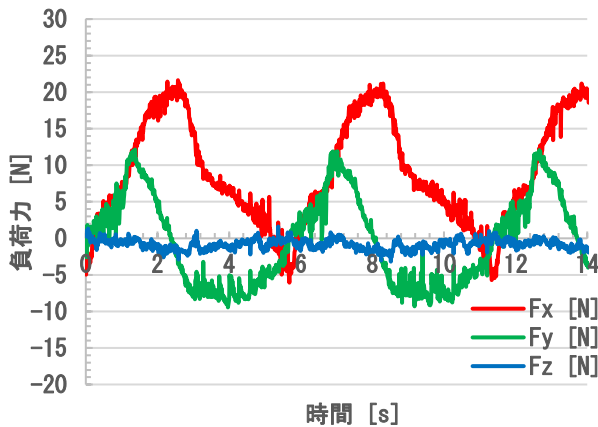


図7 バリ取り作業時の負荷力

5. 考察

5. 1. シミュレーション

穴形状を認識し、その穴の中心にロボットアーム先端をティーチングせずに移動させることができた。

また、穴の位置を変えても、その変位に応じてロボットアーム先端が追従して動作することを確認した。

今回は、画像処理として静止画を用いた。リアルタイムの画像とロボットの実機で同様の動きが再現できるかが今後の課題である。

また、穴の大きさが変わった場合、穴の個数が複数個になった場合の動作アルゴリズムの生成なども今後の課題である。

5. 2. バリ取り負荷の測定

実際に協働ロボットを用いてドリルであけた穴のバリ取り作業の負荷を実測し可視化した。

バリ取り作業時に工具先端が平面上を円運動するが、X軸方向の負荷力 F_x 、Y軸方向の負荷力 F_y が周期的に変動していることが確認できた。しかし、 F_x 、 F_y それぞれの負荷力の最大値に差があった。これらについて、工具先端を動かす動作プログラムの影響やバリの状態の違いなどが要因として考えられる。今後、詳細な検証を行っていきたい。

6. 結言

コンピュータシミュレーションにより、ティーチングの省力化の検証を行い、以下のことが実現できることを確認した。

- ・画像認識（静止画）により、穴形状を認識し、その中央にロボットアームの先端を移動させることができた。

- ・穴の位置をオフセットさせ、そのオフセットに追従してロボットアーム先端を移動させることができた。

これらを、ROSにより実現した。

今後、これらの結果を協働ロボットによるバリ取り作業に応用し、ワークの位置が数 mm から数 cm ずれたとしても、カメラで穴の位置を検出することで、ロボット動作を再教示することなく工具先端を目的の位置に移動させることができるシステムを構築する。

また、協働ロボット実機によるバリ取り作業を実施し、作業時の負荷を実測し、周期的に変動していることを確認した。バリ取り作業時の工具に係る負荷を可視化することができた。今後、バリ取り作業の仕上げとともに、この負荷がどのように変化するかなど検証を進めて行く。

参考文献

- 1) ”人口減少社会の課題と将来推計” 総務省. <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/html/nc141110.html>, (参照 2022-2-14).
- 2) ”将来人口推計シミュレーションについて” 福島県. <https://www.pref.fukushima.lg.jp/uploaded/attachment/117369.pdf>, (参照 2022-2-14).