

画像処理によるロボットアーム制御システムの開発 Development of Control System of Robot Arm Using Image Information

猿渡 太陽[†] 早坂 太一[†] 伊藤 和晃[‡]
Takaya Saruwatari Taichi Hayasaka Kazuaki Itou

麻生 優弥[§] 濱嶋 竜也[§] 城山 吉隆[§]
Yuya Asou Tatsuya Hamajima Yoshitaka Shiroyama

1. はじめに

現在、経費削減、時間短縮を目的として、産業用ロボットの研究・開発が進んでいる。しかしながら、産業用ロボットは多数の関節とリンクで構成され、複雑な動きをすることから、生産ライン等で作業させるには、ロボット付属の「ティーチングペンダント」を用いて、手でロボットを操作することによる教示、いわゆるティーチング作業を要することになる。この作業はラインを一時停止するため、生産性を低下させる上に、人件費も掛かる為、高コストであり、多くの企業にとって、省略したい作業である[1]。

現場でロボットを動かしてティーチングするのではなく、コンピュータ上でプログラムを作成し、ロボットに転送するという考え方は、一般的に「オフラインティーチング」と呼ばれる[2]。本研究では、この考え方を応用させ、カメラからの情報をコンピュータ上で処理し、ロボットに転送することで、自らの意思で対象を把持するような振舞いをさせ、ティーチング作業をなくすことを目指す。

本研究では、上記の問題を解決する手段として、無作為に配置したブロックの中から、指定した色のブロックをロボットアームに備え付けたチャックハンドで掴み、同色の他のブロックに組み付けるシステムを開発する。

2. 使用機材・開発環境

本研究では、DENSO WAVE 製の 6 軸ロボットアーム(型名: VS050A3-AV6-NNN)と OMRON 製の FQ2 センサカメラ(型名: FQ2-S30100N-08)を使用している。開発環境として、OS は Windows7, 統合開発環境は Visual Studio 2015, 言語は Visual C#, ライブラリに OpenCvSharp を使用している。

3. 基本的な画像処理

3.1 色認識

複数色のブロックのうち、指定した色のブロックを掴むために、色の認識をする必要がある。一般的なデジタル画像の色空間には RGB が使われているが、色認識には円状の領域に色相が表現された HSV 色空間が適しているため、色空間を HSV へと変換する。

色空間を変換したのちに、HSV 値の上下限を定め、範囲内の色を抽出して、マスク画像を生成する。図 1 にマスク画像の一例を示す。

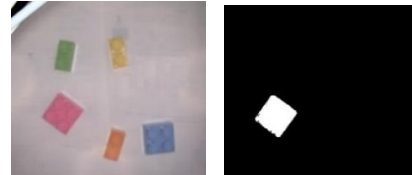


図 1 マスク画像生成

3.2 Hough 変換を用いた角度導出

的確に対象物を掴むためには、対象物の角度検出が必要である。そこで、Hough 変換アルゴリズムを用いて、画像から線分を検出する[3]。長方形のうち、得られた短辺を基準に計算することで、角度の導出が可能だが、稀に 90° 差異がある状態で角度が検出される問題が発生する。普通に Hough 変換を実行しただけでは、長辺がいくつかに分断されてしまうことにより、長辺の一部が最も短い線分と判断される事がある。これが角度誤導出の原因であると推測される。図 2 に正しい検出例を、図 3 に線分の誤検出例を示す。

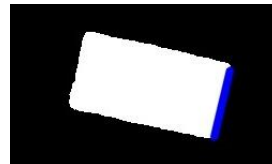


図 2 線分検出例

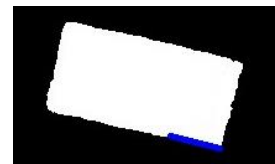


図 3 短辺誤検出例

必ず正しい線分を返す方法を考案する。まず、短辺以上の線分のみを検出するよう Hough 変換する。ここで返る線分は確実に長辺である為、この長辺を基準にした角度を返すことで、正しい角度となる。ここで長辺が一つも得られなかった場合、全線分を検出するように再び Hough 変換を行ない、得られた線分のうち平行位置にある線分同士で組を作成し、出来た線分組のうち、線分間距離が既定の長さ以上ある組は短辺を確実に検出できたと考え、短辺を基準にした角度を返すようにする。図 4 および図 5 に得られた線分組の例を示す。

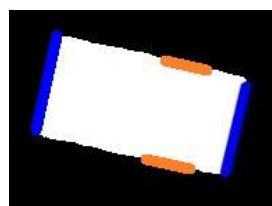


図 4 線分組の例 1

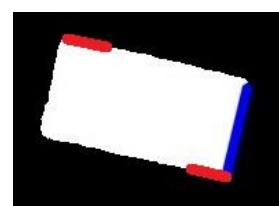


図 5 線分組の例 2

[†] 豊田工業高等専門学校, National Institute of Technology Toyota College

[‡] 岐阜大学, Gifu University

[§] 新明工業株式会社, SHINMEI INDUSTRY CO.

図 4 では、橙色と青色の線分組が検出されている。このうち橙色は既定の長さ以上の距離が無い為、棄却する。そして青色の線分組が正しい短辺と判断される。しかしながら、図 5 に示す赤色の線分組のような検出が起り得ることも想定しなくてはならないため、線分間距離だけでなく、線分の midpoint を結んだ線分の角度も考慮し、この角度が Hough 変換で得られた線分の角度のうち、いずれにも沿わなかったら棄却するようにした。ここまでの処理で、正しいと判断された線分が一つも無かった場合は、角度導出失敗とした。

4. 座標値変換

画像のピクセル値を基にした座標値と、ロボットアームの見る座標値は異なるため、カメラとロボットアームに生じる座標基準の差異を埋めるプロセスが必要である。

まず、カメラの配置位置による、カメラ中心とロボットアーム中心の差異を埋める。例えば、ロボットアームの原点座標値にあたる箇所に対象物を配置して撮影した際、得られる画像の中心に対象物が来ると容易に変換が可能だが、カメラはロボットアームの原点に位置しないために、中心差異が生じる。そこで、高さ 5[mm]ごとに中心差異を計測することで、ロボットアームの現在地とブロック頭上の差を埋めるようにした。

これらのプロセスにより、画像処理で導出した座標値とロボットの現在座標値から、ロボットの次の到達点座標値が得られる。また、高さ 5[mm]ごとに中心差異を計測したことで、多くの高さでの撮影・ロボット動作が可能であり、システムの柔軟性を向上させることが出来る。

5. ロボットアームとカメラ動作の流れ

対象物を把持するために、高さ 550[mm]地点と 350[mm]地点の 2 点の高さで各々撮影を行ない、対象物へとアプローチしていく。

流れとしては、まず高さ 550[mm]地点の座標を原点として撮影し、大まかな座標位置をとらえる。処理が完了したら、ロボットアームを対象物の真上にあたる高さ 350[mm]地点まで動作させる。その後再度撮影を行ない、この地点で角度を調整しながら対象物の把持を行なう。対象物を把持できる位置まで来たら、ハンドチャックを閉じて、実際に把持させる。その後原点へ復帰し、次は同色の大きいブロックの把持を狙う。

大きいブロックに関しても流れは同様で、原点で撮影し、高さ 350[mm]まで移動し、再度撮影し近づき、ハンドチャックを開くことで、大ブロックの頭上に小ブロックを配置するという流れである。

実際に動作させた結果、平面上にブロックを配置した場合には、正しく色認識などの処理を行ない、対象物の自動把持、自動配置が実現した。図 6 に、実際にロボットアームが対象物を把持している様子を示す。

6. 今後の課題

平面上に配置されたブロックに対しては、コンピュータ上の処理による自動把持は成功している。しかし、実際の生産ラインの現場で活かすには、図 7 に示すような山積み、斜め配置された対象物に対しての処理と自動把持が求められる。

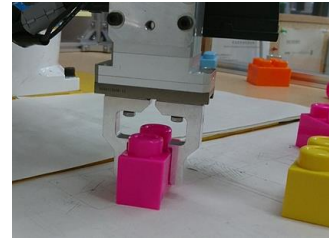


図 6 自動把持の様子

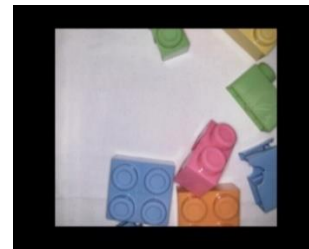


図 7 山積み配置の例

山積み配置に対して対象を把持するにあたり、ロボットアームをどの方向に傾けるか算出する為、突起面とそうでない面をそれぞれ検出する必要がある。今後はこの機能を開発しなければならない。

現在、閾値の取り方を変えることで別パターンマスク画像を生成する流れで考案している。重心座標と角度導出の場合は、図 1 に示すようなマスク画像を生成したが、突起面とそうでない面の微妙な影の違いから、一度対象物以外の情報を削除したうえでグレースケール化し、閾値を定める方法をとる。

この時、閾値は固定にするより、画像の持つ情報に応じて柔軟に変化させる方が良く考えたため、適応的二値化処理という考え方を導入する[4]。これは画像の各画素の画素値を走査し、平均値を取ることで導出できる。これにより得られたマスク画像に対して、モルフォロジー処理の一種であるオープニング処理を掛けてマスク画像を適切に加工したら、二つの面が各々の面として検出できる。そして、凸面とそうでない面を区別する手法として、円検出が適していると考えられる。

これらの処理により、ロボットアームをどの角度に振れば良いか検討がつけられる。ただ、この処理内容では、対象物の配置角度によっては、影の付き方の関係で、適切な閾値を取ることが難しく、2 面を区別できない問題がある。今後は実際にロボットアームを用いて試行すると同時に、このような問題点を解決しなければならない。

参考文献

- [1] ものづくりニュース“産業用ロボット導入が進まない訳”, <https://news.aperza.jp/産業用ロボット導入が進まない訳/>, 2018 年 6 月 14 日閲覧。
- [2] Robotmaster-Mastercam “ロボット・オフライン・ティーチングとは?”, <http://www.mastercam.co.jp/robotmaster/>, 2018 年 6 月 14 日閲覧
- [3] Qiita “やさしいハフ (Hough) 変換講座”, <http://qiita.com/YSRKEN/items/ee94c7c22599c2374722>, 2018 年 6 月 14 日閲覧
- [4] 松本 英敏 “2 値化処理法” <http://www.civil.kumamoto-u.ac.jp/matsu/kahen.pdf>, 2018 年 6 月 14 日閲覧