

卒業論文

Xtion Pro Live カメラを用いた  
複数移動ロボットのビジュアルフィードバック  
制御の基礎実験計

F112026 三木 康平

指導教員 永田寅臣 教授

2016 年 3 月

山口東京理科大学 工学部 機械工学科



概要



# 目次

第 1 章	緒言	1
第 2 章	運動学	2
2.1	順運動学 . . . . .	2
2.2	逆運動学 . . . . .	4
第 3 章	画像処理	7
3.1	二値化 . . . . .	7
3.2	Canny 法を用いたオブジェクト抽出 . . . . .	8
第 4 章	人工知能	10
4.1	人工知能 . . . . .	10
4.2	機械学習 . . . . .	11
4.3	ニューラルネットワーク . . . . .	12
第 5 章	実機による動作実験	13
5.1	Dobot . . . . .	13
5.2	学習データ . . . . .	13
5.3	実験結果 . . . . .	13
第 6 章	今後の方針	14
	謝辞	15
	参考文献	16



# 第 1 章

## 緒言

近年, 消費者嗜好の多様化により, 商品生産方式が少品種大量生産から変種変量生産へと変わってきている. それに応じて, ロボットも単一工程に特化した産業用ロボットから, 人のように状況判断を行い, 幅のある業務をこなすことができるような汎用機械としてのロボットへと, ニーズが変化してきている [1]. このニーズに応えるため, さまざまな画像処理を自動で行うカメラや, ベルトコンベア上の製品を自動で仕分けするロボットビジョンなどが販売されているが, カメラや光源の位置が限られる狭い空間内において, 対象物を画像処理により検出し移動させるシステムの実現は難しい. そこで, 研究室では狭い空間内での正確な物体の検出および移動を最終的な目標としたシステムの研究を行っている. 本研究では, 現在取り組んでいるシステムの進捗について報告する.

課題に取り組むに当たって, まず開けた空間内に平面に置かれた物体を画像認識と人工知能技術 (AI) を用いた角度推定により, 対象物の角度を依らず, 正確にピック&プレースを行うロボットシステムを開発した. この手法では, 画像内に写るオブジェクトの角度を AI で推定し, アームで Picking を行う. 具体的には, 対象物をカメラにより撮影し, まず画像認識により対象物の重心位置を計算し, その位置までアームを移動させる. その後, 角度推定用に AlexNet を基に転移学習したネットワークがその角度を推定する. そして, アームに付属するエンドエフェクタを推定角度だけ回転させ, 対象物を Picking する.

## 第 2 章

# 運動学

ロボットアームの手先を目的の位置に移動させたいとき、各関節角度と関節間の長さ現在の手先位置が分かっているならば、運動学を用いて必要な姿勢を計算することが出来る。運動学とは、ロボットアームの関節角度や関節間の長さで手先位置・姿勢の関係を数式で表したものであり、各関節角度から手先位置・姿勢を求めることを順運動学、手先位置・姿勢から各関節角度を求めることを逆運動学という。また、ロボットアームは各関節ごとに座標系（関節座標系）を設定することができ、この座標系と台座部に設定した基準座標系は運動学に用いることで互いに変換することができる。

### 2.1 順運動学

順運動学とは、ロボットアームの各関節角度の変位（回転・並進）から手先位置・姿勢を求めることであり、すなわち手先位置の関節座標系から基準座標系への変換ということができる。また、順運動学は各関節の回転と並進の要素を持つ。

#### 2.1.1 計算方法

ここでは関節座標の手先位置を基準座標系に変換する方法について示す。まず簡単のために図 2.1 のような系について考える。

移動後の原点  $O'(x'_0, y'_0, z'_0)$  は、移動前の原点  $O(x_0, y_0, z_0)$  から見ると次のように表される。

$$\begin{aligned}x'_0 &= x_0 + x_1 \\y'_0 &= y_0 + y_1 \\z'_0 &= z_0 + z_1\end{aligned}$$

上式をベクトル  $\mathbf{r}$  を用いて表すと、以下のようなになる。

$$O' = O + \mathbf{r} \tag{2.1}$$

関節座標系から基準座標系への変換を行うには、並進移動だけでなく回転移動についても考える必要がある。そこで図 2.2 のような簡単な系の回転移動について考える。



Fig.2.1. Polishing scene using a conventional industrial robot with a servo spindle.

Fig.2.2. Polishing scene using a conventional industrial robot with a servo spindle.

まずベクトル  $\mathbf{P}(x_1, y_1, z_1)$  は, 各軸方向の単位ベクトル  $(\mathbf{i}, \mathbf{j}, \mathbf{k})$  を用いて次のように表される.

$$\mathbf{P} = x_1\mathbf{i} + y_1\mathbf{j} + z_1\mathbf{k} \quad (2.2)$$

次に, Z 軸を基準としてベクトル  $\mathbf{P}$  を含む X-Y 平面を  $\theta$  だけ回転した座標系を新しく  $X', Y', Z'$  座標系とすると, その座標系におけるベクトル  $\mathbf{P}'(x_2, y_2, z_2)$  は単位ベクトル  $\mathbf{i}', \mathbf{j}', \mathbf{k}'$  を用いて次のように表される.

$$\mathbf{P}' = x_2\mathbf{i}' + y_2\mathbf{j}' + z_2\mathbf{k}' \quad (2.3)$$

ここで, 単位ベクトル  $\mathbf{i}', \mathbf{j}', \mathbf{k}'$  は回転前の座標系から見ると

$$\begin{bmatrix} \mathbf{i}' \\ \mathbf{j}' \\ \mathbf{k}' \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{i} \\ \mathbf{j} \\ \mathbf{k} \end{bmatrix} \quad (2.4)$$

と表せる. さらに式 (2.4) に式 (2.3) と式 (2.2) を代入すると

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix}$$

さらに見通しをよくするため, 単位ベクトルを省略し以下のように表す.

$$\mathbf{P}' = \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{P} \quad (2.5)$$

すなわち Z 軸を基準としたベクトルの回転移動は, 移動前後の位置移動および相対角  $\theta$  を用いて表すことができ, これは X 軸, Y 軸においても成り立つ. この式 (2.5) 中央の行列を回転行列といい  $R$  で表すこととする. 並進移動と回転移動とを用いることにより, 図 2.3 に示すような一見複雑な座標系の変換も次のような式で表すことができる.

Fig.2.3. Polishing scene using a conventional industrial robot with a servo spindle.

$${}^A\mathbf{P} = {}^A\mathbf{R}_B {}^B\mathbf{P} + {}^A\mathbf{r}_B \quad (2.6)$$

ここで  ${}^A\mathbf{R}_B$  は座標系 A から座標系 B への回転行列である. この関係を各関節ごとに用いることで, 手先位置を基準座標系からの位置として求めることができる.

## 2.2 逆運動学

順運動学が座標変換を行うことで解が 1 つ求まるのに対して, 逆運動学では図 2.4 の場合のように手先位置が同じ位置にある場合でも各リンクの位置と姿勢にはいくつかの解が存在することや, 解自体が存在しないこともある. このような問題があるため, 一般に順運動学より逆運動学の方が問題を解くのが難しい.

逆運動学でアームの関節角度を計算する例として, 図 2.5 のような場合を考える. まず, 角度  $\alpha$  と  $|\mathbf{OP}|$  の関係は余弦定理を用いて次式で表される.

$$\cos \alpha = \frac{L_1^2 + L_2^2 - |\mathbf{OP}|^2}{2L_1L_2} \quad (2.7)$$

Fig.2.4. Polishing scene using a conventional industrial robot with a servo spindle.

Fig.2.5. Polishing scene using a conventional industrial robot with a servo spindle.

式 (2.7) より角度  $\alpha$  は  $\alpha = \cos^{-1}(\cos \alpha)$  と表せるので, 結果的に  $\theta_2$  は次式で表される.

$$\theta_2 = \pi - \alpha \quad (2.8)$$

同様に, 余弦定理を用いて角度  $\beta$  と  $|OP|$  の関係は次式で表され,

$$\cos \beta = \frac{|OP|^2 + L_1^2 - L_2^2}{2|OP|L_2}$$

$\beta = \cos^{-1}(\cos \beta)$  である. 最後に角度  $\gamma$  は, 補助線  $|OP|$  と手先位置の x 座標との関係から

$$\cos \gamma = \frac{x}{|OP|}$$

であるので,  $\gamma = \cos^{-1}(\cos \gamma)$  である. よって, 角度  $\beta$  および角度  $\alpha$  を用いて  $\theta_1$  は次式で表される.

$$\theta_1 = \gamma - \beta \quad (2.9)$$

よって, 手先位置から関節角度  $\theta_1, \theta_2$  を求めることができた.

また別の方法として, 図 2.6 のような場合を考える. この場合, 手先位置は次式で表される.

$$\begin{aligned} x_p &= l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ y_p &= l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \end{aligned} \quad (2.10)$$

Fig.2.6. Polishing scene using a conventional industrial robot with a servo spindle.

式 (2.10) の両辺を時間で微分し, 行列で表すと

$$\frac{d\mathbf{P}}{dt} = \mathbf{J}(\boldsymbol{\theta}) \frac{d\boldsymbol{\theta}}{dt} \quad (2.11)$$

ここで

$$\mathbf{P} = [x_p, y_p]^T, \boldsymbol{\theta} = [\theta_1, \theta_2]^T, \mathbf{J}(\boldsymbol{\theta}) = \begin{bmatrix} -(l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2)) & -l_2 \sin(\theta_1 + \theta_2) \\ l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) & l_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$

である. 式 (2.12) は手先位置の速度と各関節における角速度の関係を表しており, 両辺の変換を担う行列  $\mathbf{J}$  を一般にヤコビ行列という. さて, 逆運動学は手先位置から各関節角度を求める問題のことであったので, 式 (2.12) を以下のように変形する.

$$\frac{d\boldsymbol{\theta}}{dt} = \mathbf{J}^{-1} \frac{d\mathbf{P}}{dt} \quad (2.12)$$

しかし,  $\mathbf{J}^{-1}$  は必ずしも存在しない.  $\mathbf{J}^{-1}$  が存在しない条件は,  $\det \mathbf{J}^{-1}$  で求めることができ, その時の姿勢を特異姿勢もしくは特異点という.

## 第 3 章

# 画像処理

画像処理とは、アナログ画像処理方式とデジタル処理方式に分けられ、前者はレンズ系や現像技術を用いて画像の特徴抽出や変換を行う手法であり、後者は画像の濃度値を画素ごとに数値化し、演算処理によって画像の特徴抽出や変換を行う手法である。今回実験に用いた手法は、後者に属するものである。この章では、まず画像内に存在する物体の重心計算手法について大まかな流れを示したあと、各段階で行っている手法について説明する。

### 3.1 二値化

まず、撮影した画像に対して二値化処理を行う。画像の二値化とは、ある閾値を定めて各ピクセルの値とその値とを比較し、画素値を白 (255) もしくは黒 (0) に置き換えることで、濃淡のある画像を二階調に変換する手法である。すなわち、閾値を  $\theta$ 、入力画像の画素値を  $I(x, y)$ 、とりうる画素値の最小値と最大値をそれぞれ 0, 255 としたとき、変換後の画素値  $I'(x, y)$  は次式で表される。

$$I'(x, y) = \begin{cases} 255 & I(x, y) \geq \theta \\ 0 & I(x, y) < \theta \end{cases} \quad (3.1)$$

この処理を行う事で、図に示すように、画像に映る物体の輪郭を取り出す事ができる。

二値化後の画像の精度は、閾値に大きく左右されるため、その設定方法に関してこれまで多くの研究がなされてきた [2, 3]。閾値の選択方法は、”大域的閾値選択法 (Global Thresholding)”と”適応的閾値選択法 (Adaptive Thresholding)”に大別される。大域的閾値処理では、画素値のヒストグラムの統計量（最大値や分布など）を用いて閾値を決定する。代表的な方法として大津の二値化法やエントロピー最大化法などがある。一方、適応的閾値選択法では、各画素の近傍領域内の統計量を利用して閾値を決定する。

本アプリケーションには、大域的閾値選択法より大津の二値化法、2 つの閾値を用いた二値化法、適応的閾値選択法より Bradley 法を実装した。

### 3.1.1 大津の二値化

大津の二値化とは、以下の式によって定義される重みづけされたクラス内分散を最小にするような閾値

### 3.1.2 Bradley 法

Bradley 法は、入力画像上の注目画素を中心として、ある領域内 ( $s \times s$ ) の画素の平均値を計算し、閾値が注目画素の画素値よりも  $t\%$  小さい場合は黒、そうでない場合は白に値を置き換える方法である。注目画素  $I(i, j)$  を中心とした領域 ( $s \times s$ ) 内の画素値の平均値は、領域内画素の総数を  $N$  とすると、次式で計算される。

$$S = \frac{1}{N} \sum_{i=\frac{s}{2}}^{i+\frac{s}{2}} \sum_{j=\frac{s}{2}}^{j+\frac{s}{2}} I(i, j) \quad (3.2)$$

二値化後の画素値を  $I'(i, j)$  とすると、この平均値  $S$  を基に、次式を用いて注目画素  $I(i, j)$  二値化を行う。

$$I'(i, j) = \begin{cases} 1 & \text{if } (I(i, j) \leq S \times \frac{100-t}{100}) \\ 0 & \text{others} \end{cases} \quad (3.3)$$

## 3.2 Canny 法を用いたオブジェクト抽出

二値画像内において、物体は画素値が等しいピクセルの集合として表現でき、その重心位置  $G(G_x, G_y)$  は、画素値とピクセル座標より式 3.4 を用いて求めることができる。

$$\begin{aligned} G_x &= \frac{\sum_{p=1}^x \sum_{q=1}^y x I(p, q)}{\sum_{p=1}^x \sum_{q=1}^y I(p, q)} \\ G_y &= \frac{\sum_{p=1}^x \sum_{q=1}^y y I(p, q)}{\sum_{p=1}^x \sum_{q=1}^y I(p, q)} \end{aligned} \quad (3.4)$$

ここで、画像の原点は左上、 $I(p, q)$  はあるピクセル座標での画素値である。この式はオブジェクトが画像中に 1 つしかない場合のみ適応可能である。しかし、オブジェクトが 1 つのみの画像は稀であり、またカメラに付着した埃などにより背景に穴ができた場合も前述の式では、重心位置がずれてしまう。そこで、背景を除き画像内でピクセル集合（面積）が最大のものを目的のオブジェクトであると仮定し、その集合に対してのみ重心位置を求めることにする。

具体的には、まずラベリング処理により、背景からオブジェクトを分離し、それぞれの面積を計算する。その後、面積が最大のものに対して重心位置を計算する。

ラベリング処理とは、2 値画像上に点在している連結成分のそれぞれに固有の名前（番号など）を付ける処理であり、これにより連結成分の個数やそれぞれの特徴（面積など）を独立して計算可能である。また、連結成分とは、図 3.1 に示すように中心画素とその周囲のいくつかの

Fig.3.1. 4 connections and 8 connections

画素が同値で存在する領域を指し、連結の種類には 4 連結と 8 連結がある。この 2 つの連結性のうち、どちらか一方を背景、他方を検出したい物体に適用することで、背景とオブジェクトの境界付近にあるピクセルの孔の影響を抑え、それらを分離することができる。本アプリケーションでは、デフォルトで 8 連結をオブジェクトに適用している。

分離された個々のオブジェクトの面積は、それに内包されるピクセル数を数えることで求めることができる。そして、面積が最も大きい集合の画像全体に対する重心位置  $G'(G'_x, G'_y)$  は次式で求められる。

$$\begin{aligned} G'_x &= \frac{\sum_{p=1}^x \sum_{q=1}^y x I'(p, q)}{S_{max}} \\ G'_y &= \frac{\sum_{p=1}^x \sum_{q=1}^y y I'(p, q)}{S_{max}} \end{aligned} \quad (3.5)$$

ここで、 $I'$  は次の条件を満たす画素値であり、 $S_{max}$  は面積の最大値である。

$$I'(i, j) = \begin{cases} 1 & \text{if } (I(i, j) \in S_{max}) \\ 0 & \text{others} \end{cases} \quad (3.6)$$

## 第 4 章

# 人工知能

### 4.1 人工知能

人工知能 (AI : Artificial IntelligenceI) は, 1947 年にアラン・チューリングによって提唱され, 現在でも活発に研究が続けられている研究分野である. しかし, その定義は正確に定められておらず, 専門家によっても次のように意見が異なる.

栗原聡	電機通信大学	人工的につくられる知能であるが, その知能のレベルは人を越えているものを想像している
山川宏	ドワンゴ人工知能研究所	計算機知能のうちで, 人間が直接・間接に設計する場合を人工知能と呼んで良いのではないかと思う
松尾豊	東京大学	人工的につくられた人間のような知能, ないしはそれをつくる技術. 人間のように知的であるとは, 「気づくことのできる」コンピュータ, つまり, データの中から特徴量を生成し現象をモデル化することのできるコンピュータという意味である

ここでは, 人工知能を「どのようなデータを, どのようなアルゴリズムで処理すれば, 人間のような知能 (認識・処理・判断など) を実現することができるのか」についての研究分野とする. また, 「AI」を実現するための様々な技術 (音声認識技術, 自然言語処理技術, 画像処理技術など) は「AI 技術」と呼ぶことにする. まず AI の歴史を図に簡単に示す. 図よりわかる通り, AI の歴史には 3 つのブームがあり, 現在はその 3 つ目にあたる.

第 1 次ブームでは, 主にコンピュータを用いた探索や推論 (ルールとゴールが決められた問題に対してコンピュータがゴールにたどり着けるよう選択肢を選んでいくもの) について研究が進められた (このような問題をトイプロブレムという). これによりパズルや迷路を解く, 数学の定理を証明する, チェスをするといった一見知的な活動が行えるようになったが, あくまでルールが網羅的かつ厳密に記述でき, ゴールが明確な問題のみ適応可能であったため, 現



実世界の問題に応用することができず、次第にブームは終息していった。

第2次ブームでは、現実問題という開いた世界を対象とし、知識中心型の研究が多く行われた。有名なものとして、エキスパートシステムが挙げられる。これは、問題領域のエキスパート（専門家）の知識をデータベースに持ち、それを利用して推論を行うことで、初心者でもエキスパートと同等レベルの判断が可能となることを目的としたものである。このシステムは現実問題を扱えることから、産業界での導入事例も多くあるが、開発されたシステムに対して、実用レベルに達したものは、極少数であった [4]。エキスパートシステムには、専門家からヒアリングによって得た知識をコンピュータが理解できる表現に変換することが困難なこと、実際にシステムで利用できるよう知識が少ないこと、そしてエキスパートシステムの整理された知識は一般的に汎用化することが難しいことなどの問題 [4, 5] があり、次第に知識中心型の AI の研究は下火になっていった。

しかし、

## 4.2 機械学習

コンピュータは、計算のルールや値の格納場所を明示的にプログラムすることによって、入力を自分が望む出力に変換する。そのため、ルールが決められないような問題を解くことを苦手としてきた。そこで、「ルールの記述が難しい問題（例えば1枚の画像から人の顔を見分ける問題）」に対して、人間がルールを決めるのではなく、コンピュータ自体にルールを発見させようとする手法が機械学習である。また、機械学習の一般的な説明には、以下の A.L.Samuel によるインタビューがよく用いられる [9]。

“The field of study that gives computers the ability to learn without being explicitly programmed. (明示的にプログラミングすることなく、コンピュータに学ぶ能力を与えようとする研究分野).”

機械学習の分類は、神畠、鹿島による同論文に従うことにする。

栗原聡	電機通信大学	人工的につくられる知能であるが、その知能のレベルは人を越えているものを想像している
山川宏	ドワンゴ人工知能研究所	計算機知能のうちで、人間が直接・間接に設計する場合を人工知能と呼んで良いのではないかと 思う
松尾豊	東京大学	人工的につくられた人間のような知能、ないしはそれをつくる技術。人間のように知的であるとは、「気づくことのできる」コンピュータ、つまり、データの中から特徴量を生成し現象をモデル化することのできるコンピュータという意味である

## 4.3 ニューラルネットワーク

## 第 5 章

# 実機による動作実験

### 5.1 Dobot

TechShare 社が販売している Dobot Magician(Dobot) は, 4 自由度のロボットアームでありエンドエフェクタを取り換えることによって 3D プリンタのような積載加工, エンドミルによる切削加工ペンツールを使った印字などが行える. Dobot の質量と可搬質量はそれぞれ 3.4kg と 500g であり, 位置繰り返し精度は 0.2mm と教育ロボットとしては優れている. また Dobot 社より, C や MFC(Microsoft Foundation Class), Python など複数のプログラミング言語で API が提供されており, 個人でプログラムを組む際の敷居も低い. 今回は, その中の Python API を統合開発環境である Visual Studio2019 上に実装し, Dobot 本体を動作させるプログラムの開発を行った. API とはあるコンピュータプログラム (ソフトウェア) の機能や管理するデータなどを, 外部の他のプログラムから呼び出して利用するための手順やデータ形式などを定めた規約のことである. 今回はその中から Dobot 本体の姿勢を取得するための `GetPose()` また, Python に標準で搭載されている GUI 作成キットである Tkinter を用いて, 開発した動作プログラムを簡単に扱えるよう Dobot 制御用のユーザインタフェースの作成も行った.

### 5.2 学習データ

### 5.3 実験結果

## 第 6 章

### 今後の方針

## 謝辞

本研究は，山口東京理科大学大学院基礎工学研究科基礎工学専攻で行われたものである．

## 参考文献

- [1] 井尻善久, F. Drigalski, “産業用ロボットの進化によるものづくりの近未来”, 日本ロボット学会誌, Vol. 37, No. 8, pp. 5-8, 2019.
- [2] N. Otsu, “A Threshold Selection Method from Gray-Level Histograms.”, IEEE Transactions on Systems. Man. and Cybernetics, Vol. 9, No. 1, pp. 62?66, 1979.
- [3] D. Bradley, G. Roth, “Adapting Thresholding Using the Integral Image”, Journal of Graphics Tools, Vol. 12, No. 2, pp.13?21, 2007.
- [4] 寺野隆雄, “エキスパートシステムはどうなったか?”, 計測と制御, Vol. 42, No. 6, pp. 458-462, 2003.
- [5] 諏訪基, 小林重信, 岩下安男, 國藤進, “エキスパートシステム開発事例にみる知識獲得の諸相”, 計測と制御, Vol. 25, No. 9, pp. 31-39, 1986.
- [6] 新井紀子 『AI vs. 教科書が読めない子供たち』(東洋経済新報社, 2018)
- [7] 国立研究開発法人科学技術振興機構研究開発戦略センター: (研究開発の俯瞰報告書) システム・情報科学技術分野 (2019), <https://www.jst.go.jp/crds/report/report02/CRDS-FY2018-FR-02.html>
- [8] 合原一幸 『人工知能はこうして創られる』(株式会社ウェッジ, 2017)
- [9] 神畠敏弘, 鹿島久嗣, “機械学習分野の俯瞰と展望”, 人工知能学会誌, Vol. 34, No. 6, pp. 905-915, 2019.