

2024年度 卒業論文

人のような描き方ができる頭部の
線画生成の研究

2024年3月

指導教員

藤江真也 教授

千葉工業大学 先進工学部
未来ロボティクス学科

森田大雅

目次

第1章	序論	1
1.1	背景	1
1.2	本研究の目的	2
1.3	本論文の構成	2
第2章	関連研究	3
2.1	描画ロボットに関する研究	3
2.2	エッジ抽出に関する研究	3
第3章	システム構成	5
3.1	ハードウェア	5
3.2	ロボットの機構	5
3.3	手先の位置と回転角の関係	5
3.4	画像処理	8
第4章	提案手法	10
4.1	描き順の分析	10
4.2	領域と端点を用いた経路の求め方	10
4.2.1	スタート地点の決め方	10
4.2.2	端点の検出と線の辿り方	11
第5章	実験	12
5.1	概要	12
5.1.1	ラスタスキャン	12

第 6 章 結果	13
6.1 描画の結果	13
第 7 章 結論	14
謝辞	15

図 目 次

3.1	リンク座標系 (仮)	6
3.2	エッジ抽出	8
4.1	領域の位置の確認	11

表 目 次

第1章 序論

1.1 背景

現代ではデジタルで絵を描いたり、AI が絵を生成するようになっている。そのため、アナログで描くことにこれから意味があるのか疑問に思う人もいると思う。コンピュータを使って自分の表現したい色や絵を生成することよりも、絵画のような人が数多くの鍛錬を積み、美しい絵や不思議と魅力あふれる絵を自力で描いていくことは、これからも一定の価値を持ち続けると考えられる。しかし描く手段が増えれば、効率化のためにデジタルを用いることが増える可能性は考えられる。

アナログの場合、たとえ見た目が同じに見えても絵の具を使った量や混ぜた色の比率、筆のタッチの刻み方や手のぶれなどを完全に再現できることは不可能であるため、優れた絵描きの絵のオリジナリティは強く評価される。また時間と労力は有限であるため、作品数は限られるため希少価値は高くなる。

このように人がアナログ絵を描く価値はこれからもあり続けると考えられるが、ロボットがアナログ絵を描いた場合はどうなるだろうか。私は、簡易的な絵であれば、ロボットに描いてもらいたい場合が存在すると考える。例えば、ミニ黒板を用いたカフェや飲食店などの看板イラストや、地域のイベントのポスターイラストなどである。地域のイベントでアイデアを募集するもの良いが、このようなときに簡単なイラストが描けるロボットが存在し、活用できれば時間とコストの削減が期待できる。

1.2 本研究の目的

従来の描画ロボットの研究では描かれた絵が、どのくらい上手く描けているかに着目している研究が多い。例えばシンプルにエッジ抽出から人物画を描く文献 [1] や、エッジ抽出とハッチングから芸術的な人物画を鉛筆で描く文献 [2]、リモートユーザがタブレットを介してロボットに描かせる文献 [3] などが存在する。これらの研究では模写や芸術的な表現が可能になっているが、人のような描き方を追求したものが少ないと感じた。

また、風景や静物などは描き手によってかなり描き順に差があるが、人物画の頭部であればある程度パターンがあると考えた。加えて、ヒトの頭部は表情が存在するため絵に様々な意味を連想させたり、静止した一場面にストーリーをもたせたり、背景などの他の要素をより際立たせることができる。これは絵画以外の漫画にも当てはまることで、頭部を描く重要性は高い。そこで本研究では、ある画家が描いた頭部作品の画像から描き順ができるだけ人に近い描き方をする描画ロボットを作成を行う。

1.3 本論文の構成

第2章 関連研究

描画ロボットの研究はあまり多くないが、マニピュレータを使った先行研究とよく用いられるエッジ検出を行うアルゴリズムに改良を加えた研究を調査した。本研究では線画を描くためにエッジ抽出を用いている。画像からエッジ抽出を行うとノイズがどうしてもものってしまうため、ノイズをできるだけ低減できる文献がないか調査した。

2.1 描画ロボットに関する研究

文献 [1] では、6 軸のマニピュレータロボットを用いて、シンプルにエッジ抽出から人物画を描く研究を行っているのに対し、文献 [2] は Canny やラプラシアンなど様々なエッジ抽出と細線化、そして影の部分をハッチングして芸術的な人物画を描く研究を行っている。また文献 [3] は、リモートユーザがタブレットを介してロボットに描かせる研究を行っている。これらの研究では模写や芸術的表現が可能になっており、絵描きが描いたようなとても質の高い絵が描かれていた。

2.2 エッジ抽出に関する研究

用途にもよるがエッジ抽出において、Canny のエッジ検出アルゴリズムが最もよく用いられる。エッジ抽出を行うとき、しきい値を高くするとノイズは少なくなるが、エッジが見えなくなってしまう。逆にしきい値を低くするとエッジは残るが、ノイズがはっきりと現れてしまう。これらのトレードオフ関係に取り組んだ論文が文献 [4] である。この文献は画像を領域ごとに仕切り分けを行い、その領域ごとにエッジがあるのか、ノイズがあるのか、両方あるのかを判定し、領域ごとに

しきい値の値を決め、トレードオフ関係に対処しようとしている。結果的にオリジナルより、少し性能が向上したことが示されている。

第3章 システム構成

3.1 ハードウェア

ロボットの機体は Adept の ADA031 を用いている. Arduino で動作するため、画像処理の難易度が高い. そこでメインボードを Raspberry Pi に代え、それとロボットアームの各 servo モータを接続するためにブレッドボードやジャンパー線、抵抗を用いて改良を行った.

3.2 ロボットの機構

本研究で用いるロボットは 3 軸のマニピュレータロボットである. servo1 を原点とし、各 servo の回転角度を $\theta_1, \theta_2, \theta_3$ 、各リンクの長さを l_1, l_2, l_3, l_4 とする. 図 3.1 はリンク座標系の定義と、実機の様子を示したものである.

3.3 手先の位置と回転角の関係

リンク座標系に定義した各パラメータと、順運動学と逆運動学を用いて手先の位置から回転角を求める. また DH(Denavit-Hartenberg) 記法を用いて、各リンクの関係を示す. このリンク座標系をもとに、座標系 1 から 3 への座標変換行列を求

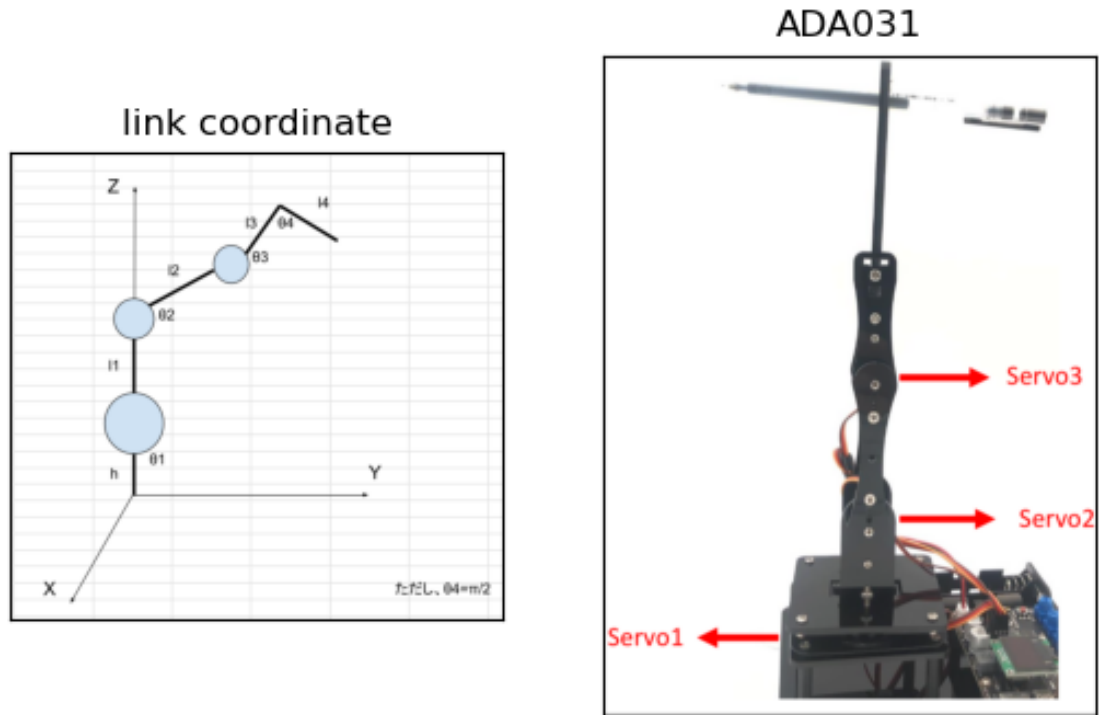


図 3.1: リンク座標系 (仮)

めると以下ようになる.

$${}^0T_3 = {}^0T_1 {}^1T_2 {}^2T_3 = \begin{pmatrix} C_1 C_{23} & -C_1 S_{23} & 0 & l_2 C_1 S_2 \\ S_1 C_{23} & -S_1 S_{23} & 0 & l_2 S_1 S_2 \\ -S_{23} & -C_{23} & -1 & l_2 C_2 + l_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.1)$$

ただし、

$$\begin{pmatrix} C_{23} = \cos\theta_2 \cos\theta_3 - \sin\theta_2 \sin\theta_3 \\ S_{23} = \sin\theta_2 \cos\theta_3 + \cos\theta_2 \sin\theta_3 \end{pmatrix}$$

また、手先ベクトルが以下のように求まる.

$$\begin{pmatrix} 1 & 0 & 0 & l_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & l_4 \\ 0 & 1 & 0 & -l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.2)$$

式 (3.1) と (3.2) より手先までの座標変換行列 0P_r が以下のように求まる.

$${}^0P_r = {}^0T_3 \begin{pmatrix} 1 & 0 & 0 & l_4 \\ 0 & 1 & 0 & -l_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix}$$

ただし、回転行列 \mathbf{R} と並進ベクトル \mathbf{t} は以下の結果になる.

$$\begin{aligned} \mathbf{R} &= \begin{pmatrix} C_1C_{23} & -C_1S_{23} & 0 \\ S_1C_{23} & -S_1S_{23} & 0 \\ -S_{23} & -C_{23} & -1 \end{pmatrix} \\ \mathbf{t} &= \begin{pmatrix} l_4C_1C_{23} + l_3C_1S_{23} + l_2C_1S_2 \\ l_4S_1C_{23} + l_3S_1S_{23} + l_2S_1S_2 \\ -l_4S_{23} + l_3C_{23} + l_2C_2 + l_1 \end{pmatrix} \\ \mathbf{0} &= \begin{pmatrix} 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

座標変換行列の並進ベクトル \mathbf{t} が順運動学解なので、手先の位置 x, y, z が以下のように求まる.

$$\begin{aligned} x &= C_1(l_4C_{23} + l_3S_{23} + l_2S_2) \\ y &= S_1(l_4C_{23} + l_3S_{23} + l_2S_2) \\ z - l_1 &= -l_4S_{23} + l_3C_{23} + l_2C_2 \end{aligned} \quad (3.3)$$

式 (3.3) より逆運動学解 $\theta_1, \theta_2, \theta_3$ は以下のように求まる.

$$\begin{aligned}\theta_1 &= \frac{1}{2} \cos^{-1} \left(\frac{x^2 - y^2}{x^2 + y^2} \right) \\ \theta_2 &= \cos^{-1} \left(\frac{x^2 + y^2 + (z - l_1)^2 + l_2^2 - l_3^2 - l_4^2}{2l_2 \sqrt{x^2 + y^2 + (z - l_1)^2}} \right) + \tan^{-1} \left(\frac{\sqrt{x^2 + y^2}}{z - l_1} \right) \\ \theta_3 &= \cos^{-1} \left(\frac{x^2 + y^2 + (z - l_1)^2 - l_4^2 - l_3^2 - l_2^2}{2l_2 \sqrt{l_3^2 + l_4^2}} \right) + \tan^{-1} \left(\frac{-l_4}{l_3} \right)\end{aligned} \quad (3.4)$$

3.4 画像処理

描く対象の絵は、ウィリアム・ブーグローという画家の”若い女の横顔“という作品の画像を用いる. 理由は背景が白でシンプルであるため、画像の読み込む際にノイズが少なく綺麗にエッジ抽出を行えるを考えたからである. この画像からエッジ抽出を行い、線画を生成する. エッジ抽出には OpenCV の Canny のエッジ検出アルゴリズムとガウシアンラプラシアンに細線化を行ったものの2つを比較し、Canny を使うことに決めた. 以下のそれぞれのエッジ抽出の結果を示す.

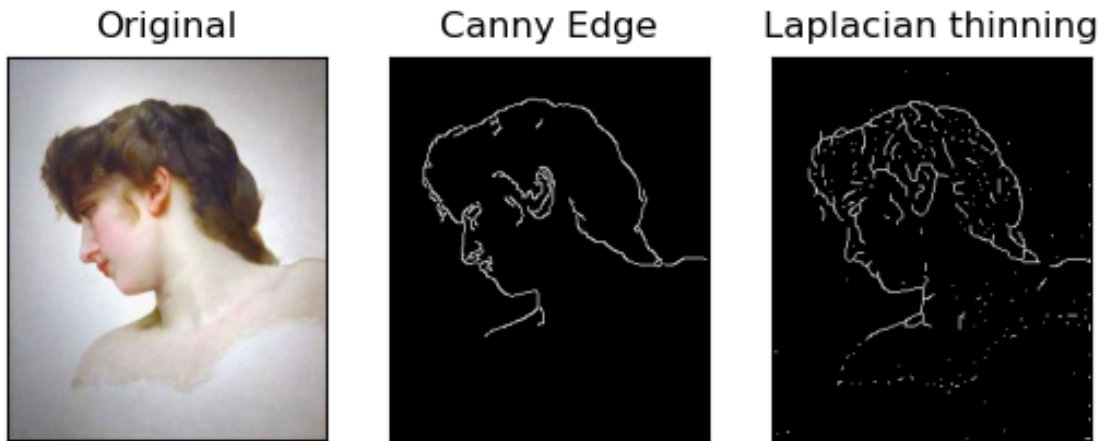


図 3.2: エッジ抽出

図の 3.1 からラプラシアンフィルタより、Canny の方ががノイズが少なく綺麗に

エッジ抽出できていることが分かる。そのため、Canny でエッジ抽出を行った結果を線画とする。この出力画像をもとに線の経路を求めていく。

第4章 提案手法

エッジ抽出を行った画像から線の経路を求める。できるだけヒトが描いたような描き順になるようにするために、まず実際にこの絵を自分が描いたときの描き順を調べた。右利きと仮定すると、左側から描き始めることが多かった。そのため、本研究ではスタートを頭部の輪郭の左側から描けるようにした。そのためにある領域を端点を用意する。

4.1 描き順の分析

私は顔の輪郭から始まり首や目、鼻などの細部へと移って描いていく。これは全体から細部へと移ることで、パーツの位置を決めやすい体と考えられる。また右利きの場合、スタートは左側から描いていた。理由は右利きの場合、ペンを右に傾けて持つため時計周りに円を描くほうが描きやすいからだと考えられる。半時計周りの場合、6時から12時までの区間を描くには手を少し持ち上げて描く必要があるため、寝かせたまま描ける時計周りより不安定になる。

4.2 領域と端点を用いた経路の求め方

4.2.1 スタート地点の決め方

スタート位置を決めるためにある大きさの領域を用意した。領域は左上の位置が横全体の $\frac{1}{8}$ 、縦全体の $\frac{1}{3}$ 、サイズが全体の $\frac{1}{4}$ になるように設定した。このように領域の位置とサイズを定義することで、頭部の左側に領域が被るように設定することができた。領域が頭部の左側にあるかを2つの画像で確かめた。図4.1の左が”金髪女の横顔”、右が”若い女の横顔”という作品である。以下にその例を示す。

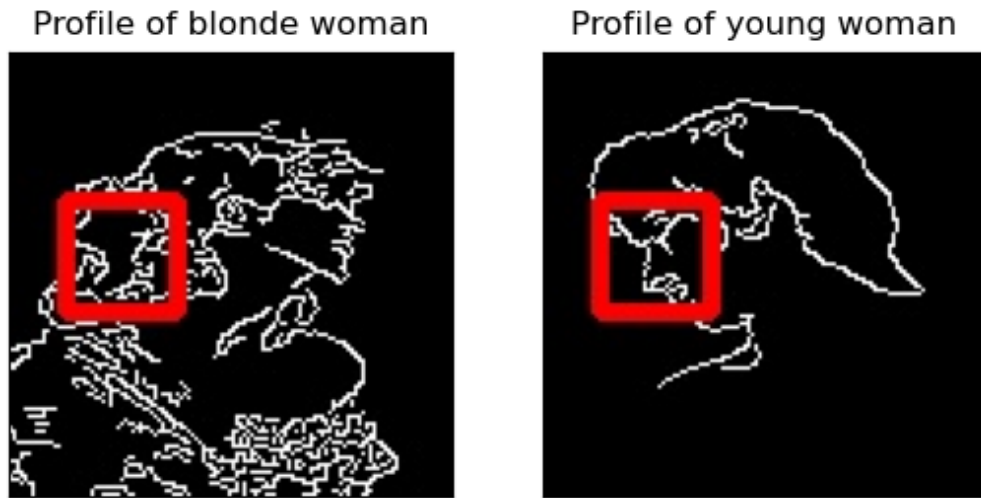


図 4.1: 領域の位置の確認

この領域を縦方向にスキャンし、最初にエッジの画素にぶつかった位置をスタート地点に決める。

4.2.2 端点の検出と線の辿り方

図 4.1 ではエッジが綺麗に繋がっているように見えるが、拡大すると途切れている。そのため、ロボットにそのままエッジの画素をそのまま座標として与えると実際の線画が点線になってしまう。そこで端点を用いる。線の画素を辿り、端点まで来たら次に近くの端点へ移動して描いていく。この処理を行うことで実際には点線を描くことになるが、擬似的に線を描いていくような動作ができる。線の辿り方はある線の画素から隣に線の画素があるかを探し、線の画素があれば移動し、なければ最も近いまだ通過していない端点へと移動する。これを線の画素がなくなるまで繰り返す。

第5章 実験

5.1 概要

経路の求めるためにラスタスキャンと領域、端点を用いた方法のどちらが人のような描き方に近いかを比較し、どちらの描き順が人が描いたように見えるのかアンケートを 100 人にとった。

5.1.1 ラスタスキャン

ラスタスキャンとは、左上から右下へ向かってたどる方法である。右端にたどり着いたら一段降りてまた左端から右端へたどる。この処理を用いて線の画素を探す。線の画素を見つけたら、端点にたどり着くまで移動し続ける。端点に辿りついたら、もとの位置に戻ってまたスキャンを再開する。これを線の画素がなくなるまで繰り返す。

第6章 結果

〇〇人にラスタスキャンと領域、端点を用いた方法のどちらが人のような描き方に近いアンケートをとった結果、〇〇人が〇〇と答えた.

6.1 描画の結果

以下に今回用いたロボットに描画させた絵の画像を示す.

第7章 結論

謝辭

参考文献

- [1] M. Pichkalev, R. Lavrenov, R. Safin and K. -H. Hsia, "Face Drawing by KUKA 6 Axis Robot Manipulator," 2019 12th International Conference on Developments in eSystems Engineering (DeSE), Kazan, Russia, 2019, pp. 709-714, doi: 10.1109/DeSE.2019.00132.
- [2] Felix Fisgus, Joris Wegner: "Pankraz Piktograph".
- [3] Shubham Agarwal, Sarvesh S. S. Rawat, V Sumathi: "A drawing robotic hand based on inverse kinematics", International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, India, 2014, pp. 1-5, doi: 10.1109/ICICES.2014.7034005.
- [4] D. Dhillon and R. Chouhan, "Enhanced Edge Detection Using SR-Guided Threshold Maneuvering and Window Mapping: Handling Broken Edges and Noisy Structures in Canny Edges," in IEEE Access, vol. 10, pp. 11191-11205, 2022, doi: 10.1109/ACCESS.2022.3145428.
- [5] 広瀬 茂男 著 機械工学選書 裳華房 ロボット工学-機械システムのベクトル解析-
- [6] 細田 耕著 実践ロボット制御-基礎から動力学まで-
- [7] デジタル画像処理 (改訂第二版) 発行所 CG-ARTS (公益財団法人 画像情報教育振興協会)