

ロボットアームを用いた カメラの位置制御による視野拡大に関する研究

諏訪 航太郎 大川 猛

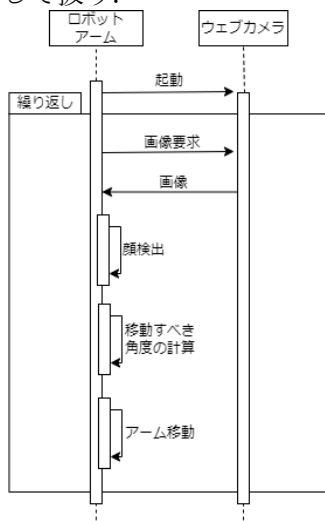
東海大学情報通信学部組込みソフトウェア工学科

1. はじめに

新型コロナウイルスの流行を経て検温の機会が増加している[1]. しかし, 測定者が必要な手持ちの検温器には接触による感染リスクがあり, スタンド型の検温器にはサーモカメラの画角の関係で検温を終えることが出来ない場合がある. 本研究では, ウェブカメラとロボットアームを用いてカメラの位置を制御し, カメラの画角の中心に対象の顔を捉え続けることで, カメラの視野拡大を行うことを目的とする.

2. 視野拡大のためのシステムの設計

顔検出結果に基づき, カメラ位置を制御することで, 視野拡大を行うシステムを設計した. 顔検出のために, サーモカメラとは別のウェブカメラを利用し, そこから手に入れることができる映像を用いる. 検出した顔の座標に対しロボットアームを動作させ, ウェブカメラとサーモカメラの位置を制御することによってその視野拡大を実現する. システムのシーケンス図を図1に示す. 本研究では計算機を内蔵したロボットアームを利用したため, 図1では計算機とロボットアームを同一のものとして扱う.



【図1】 視野拡大システムのシーケンス図

ロボットアームの関節角度を制御するためのアルゴリズムは以下の通りである.

$$\alpha = X - (x + w/2) * \text{cam_rad} / 360 \quad \text{式(1)}$$

$$\beta = Y - (y + h/2) * \text{cam_rad} / 360 \quad \text{式(2)}$$

α = 移動すべき角度(縦) β = 移動すべき角度(横)

X = 解像度(縦)

Y = 解像度(横)

x = 顔の位置(縦)

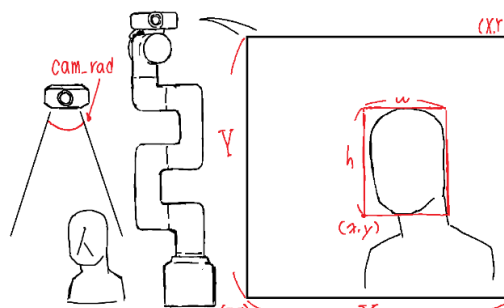
y = 顔の位置(横)

w = 顔の幅

h = 顔の高さ

cam_rad = カメラの画角

式(1), (2)について補足する図2を以下に示す.



【図2】 視野拡大システムの概念図

3. 実験

実験として, 小型コンピュータを搭載したロボットアームを用いて視野拡大システムを開発し, 動作時間の計測を行った. ロボットアームとして, Raspberry Pi 4 Model B を内蔵する myCobot 280-pi [2] を使用した. ロボットアームの先端にはウェブカメラ (BSWH06M, バッファロー社製 120万画素) を設置した. ウェブカメラから取得した画像からの顔検出には OpenCV の Haar-Like 特徴量を用いたカスケード分類器 [3] を使用した. 移動すべき角度の計算を式(1), (2)で行い, アーム移動には pymycobot を使用した. これは本実験で利用したロボットアーム myCobot 280-pi を動作させるためのライブラリである.

表1に, 視野拡大システムの動作時間計測結果を示す. 画像の解像度は, 320×240 と, 640×480 の二種類について, 図1に示した処理ステップ毎に計測を行った. 解像度 640×480 の場合, 顔検出の時間が 182ms と時間がかかる事が分かった. また, 角度取得とアーム移動は pymycobot ライブラリの API を使用しており, センサ・アクチュエータの入出力に 70-80ms を要することが分かった.

Research on field of view expansion by controlling camera position using a robotic arm
Kotaro Suwa, Takeshi Ohkawa
Department of Embedded Technology, School of Information and Telecommunication Engineering, Tokai University

【表1】視野拡大システムの動作時間計測結果

処理区間	解像度と平均処理時間[ms]	
	320×240	640×480
画像取得	7.71	14.47
顔検出	58.61	182.41
角度取得	67.69	75.43
角度計算	0.07	0.04
アーム移動	79.40	80.78

4. 考察

本論文で設計した視野拡大システムの動作時間計測結果を基に、計測対象がウェブカメラの画角の端から正面に来るまでに顔検出を行う事が出来る計測対象の最大速度を計算した。計算の際に顔検出の射程を2mと仮定し、ウェブカメラの画角は実験で使用したものと同じ±31.5°とした。また、顔検出の精度は十分にあり、計測対象はカメラの正面にに対して垂直に横切るものとしている。使用した式は以下の通りである。

$$V = (l * \sin \theta) * 1000 / (a + b + c + d + e) \quad \text{式(3)}$$

V = 対象を検出可能な最大速度

l = 顔検出の射程 θ = 正面から画角の端までの角度

a = 画像取得時間 b = 顔検出時間 c = 角度取得時間

d = 角度計算時間 e = アーム移動時間

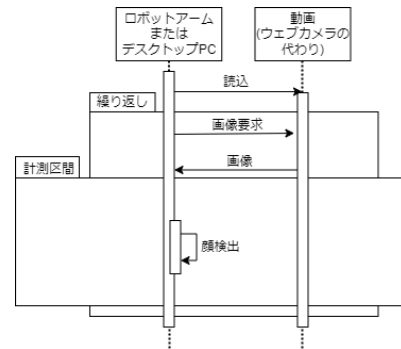
計測結果を表2に示す。

【表2】解像度と計測できる速度の対応

解像度	320×240	640×480
検出可能な最大速度	16.08km/h	10.66km/h

表2の計算結果から、検出対象からの顔検出には十分な処理時間であるといえる。しかし、実際には本論文で用いた顔検出では横顔を検出することができず、近くを通られることも予測できる為、検出可能な最大速度は下がると考えられる。

表1に示すように、視野拡大システムでは顔検出、角度取得、アーム移動に処理時間が多くかかっている事が分かった。そこで、処理時間のかかる処理の内、処理時間短縮の見込みがある顔検出について、別のより高性能な計算機に顔検出処理を行わせることが処理時間の短縮につながるかについて調査を行った。調査のために計測用システムとして、ウェブカメラから画像を取得する代わりに予め用意した動画を1フレームずつ読み込んで顔検出を行うシステムを作成した。動画を読み込ませることで、計測対象の違いが結果へ影響することを防止した。動画は解像度 360*640 のものを使用した。このシステムをロボットアームとより高性能な計算機として用意したデスクトップ PC に実装し、それぞれで顔検出部分の処理時間を測定した。計測には time モジュールを使用した。計測区間を表すために、図3にシーケンス図を示す。



【図3】計測用システムのシーケンス図

【表3】計測用システムの動作時間測定結果

使用機材	平均処理時間[ms]
myCobot 280-pi	約 2200
デスクトップ PC (Xeon E3-1275 3.8GHz)	21.15

計測用システムをロボットアーム、デスクトップ PC のそれぞれで動作させた際に顔検出にかかった時間を表3に示す。

表3に示すように、ロボットアームで測定した処理時間の約 2200ms に対し、デスクトップ PC で測定した処理時間は 21ms である。2つの計測結果には 100 倍超の差が存在するため、デスクトップへの画像通信を介した場合に全体の処理時間の短縮につながることを期待できる。しかし、表1での解像度 640*480 と表3での解像度 360*640 ではピクセル数が 0.75 倍であるのに対して、処理時間が 182ms から約 2200ms までおよそ 12 倍に増えているので、ロボットアームでの測定で得られた処理時間は別の原因による遅延が存在する可能性が考えられる。

5. まとめ

本研究ではロボットアームを用いたカメラの位置制御による視野拡大を行うシステムを開発し、その動作時間について測定した。また、顔検出が有効な対象の速度と画像処理部分の処理時間短縮のための考察を行った。結論として、限られた条件の中で目的を達成できることが分かったが、処理速度について課題も示された。

6. 参考文献

[1] 体温計に関する意識・実態調査 2021
タニタ調べ

[2] Elephant Robotics myCobot 280-pi
<https://www.elephantrobotics.com/en/mycobot-pi/>

[3] Paul Viola and Michael J. Jones. Robust real-time face detection. International Journal of Computer Vision, 57(2):137-154, 2004.