

# マルチスレッドと Coarse-to-Fine 法を用いたステレオビジョンカメラの開発

## Development of stereo vision camera using multithread and Coarse-to-Fine method

1225064 大西 林吾 (Soft Intelligent System on Chip 研究室)  
(指導教員 星野 孝総 准教授)

### 1. はじめに

ROBO-剣と言われるロボット競技がある。この競技はアーム型ロボットによる剣道大会であり、関節型ロボットの基本を学ぶことや、画像処理や人工知能の技術育成を目指している。2019年時点で第10回が開催されている。ルールは人間の剣道大会と同じようにロボットが相手を竹刀で叩くことでポイントを得ることができる。出場するロボットは人間の面、胴、小手にあたる部分が赤、青、黄に色分けされておりカメラによる認識がしやすいようになっている。部門は手動制御と自律制御に分かれているが本研究では、自律制御でROBO-剣へ出場することを研究の最終目標とした。ロボットが自律制御を行うためには様々な技術が必要である。ROBO-剣へ出場することを考えると入力装置であるカメラの開発は必須であると考えている。そこで本研究では相手ロボットの認識のためにUSBカメラを用いてステレオカメラを作成した。本研究の目的は作成した距離推定システムの処理速度を向上させることである。そのためにCoarse-to-Fine法とマルチスレッドを用いて高速化を試みた。

### 3. Coarse-to-Fine 法

画像処理の分野では計算量を削減するときにCoarse-to-Fine (粗密)法と言われる方法を用いる場合がある。仕組みとしては解像度の異なる画像群 (画像ピラミッド) を用意し段階的に処理する方法である。本研究のUSBカメラを用いた距離推定システムでCoarse-to-Fine法を用いることを考える。USBカメラから得られた画像を可能な限り小さくリサイズしてラッキングを行う。画像サイズが小さいままだと分解能が下がるため、ラッキングして得られた座標をもとにオリジナルの解像度の画像から対象の領域のみをトリミングする。そして、トリミングした画像に対してラッキングを行うことで、計算量を減らしつつ解像度の高い画像に対して処理を行うことが可能となる。画像処理の手順としては複雑になるが計算量は削減することができると考えられる。

### 4. マルチスレッド

プログラムの高速化を考えたときに並列処理によってプログラムを高速化する手法が用いられることが多い。近年のCPUはマルチコア化が進んでおり、コアごとにタスクを分割することでリソースを無駄なく使用しプログラムを高速に処理することが可能である。図1に本研究で設計した距離推定システムのスレッド分割について示す。スレッド1はメインのスレッドであり画像の描画、距離の計算、時間の計測を行っている。スレッド2は左USBカメラからの画像を取得し対象の重心位置の取得までを行う。取得された重心位置は共有メモリ上の変数に格納される。スレッド3はスレッド2と同様の処理を右USBカメラからの画像に対して行う。

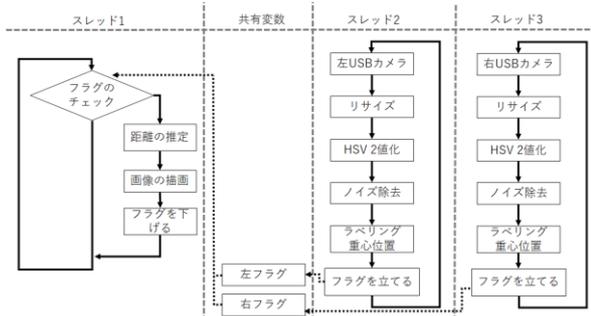


図1. マルチスレッドの設計

### 5. 実験

#### 5.1 実験概要

使用したパソコンのスペックは、CPUはi5-2400Sでありメモリは8GByteである。USBカメラには一般的なwebカメラとして使用される30fpsのUSBカメラを用いた。このUSBカメラは1280×960のサイズで画像を取得することができる。比較した条件は、ステレオカメラから画像の取得のみのときと、従来の方法 (シングルスレッド) で距離推定を行ったとき、マルチスレッドを用いた場合、Coarse-to-Fine法を用いたときと比較を行った。また、マルチスレッドとCoarse-to-Fine法を用いた場合を統合実験として行った。時間の計測は、画像の取得から距離の計算が終了したタイミングを1回と数え500回のループの平均の処理速度とした。マルチスレッドの場合はスレッド1の時間を計測した。

#### 5.2 実験結果



図2. Coarse-to-Fine法を用いた色抽出と重心位置の描画  
表1. 実験結果 (単位: ms)

	描画のみ	1色のみ	3色同時
従来 (シングルスレッド)	33.33	51.34	104.02
フィルタ改善		66.60	67.27
Coarse-to-Fine		47.70	73.60
マルチスレッド	33.51	35.10	65.88
統合実験		33.39	34.54

図2にCoarse-to-Fine法を用いて距離推定を行ったときモニタのキャプチャ画像である。距離の推定精度については正しい計測をしていないが概ね正確に行うことを確認した。図2の左側には左USBカメラから得られた画像に、赤、青、黄のそれぞれの重心位置に点が描画されていることが確認できる。右側にはリサイズした画像をもとに得られたそれぞれの色の重心位置の座標をもとにオリジナルの解像度の画像からトリミングされた領域を描画している。表1に処理速度の比較を行った実験結果を示す。処理速度は従来の3色同時に距離推定を行ったときと比較すると37.5%の処理速度の工場となった。

### 6. まとめ

USBカメラを用いた距離推定システムにおいて、Coarse-to-Fine法とマルチスレッドを用いることで非同期的なリアルタイムラッキングと距離の推定を行うことが可能であることを示した。今後の取り組みとして算出された相手ロボットの色と位置情報をもとに、どのようにロボットを制御するか考える必要がある。