

床指紋を用いた室内位置推定の高速化

藤田 貴大[†] 藤田 悟[†]

法政大学情報科学部[†]

1. 序論

近年、位置情報を用いたサービス、アプリが普及してきている。この用途のために、利用者の位置推定は必要不可欠な技術であり、屋外では一般的にGPSが利用されている。一方、GPSによる位置測位が困難な屋内では、GPSの代替技術が検討されてきた。しかし、未だに決定的な手法が確立されていない。屋内の位置推定手法に、場所による床面の模様、傷、汚れの違いを判別することで撮影場所を推定する床指紋という手法がある。この床指紋を用いて実際に位置推定が実現可能であるが、照合にかかる実行時間の問題などがあつた。そこで本論文では、床指紋の照合の実行時間短縮を目指す。

2. 床指紋について

床指紋の照合は、スマートフォンで撮影された床画像の特徴点をマッチングすることで実現する[2]。特徴点抽出と特徴点マッチングはAKAZEを使用した。床指紋では、床面の微細な傷や汚れなどの特徴を抽出しているため、AKAZEでのマッチングペアの大多数は正しい位置の対応点で結ばれていないペアである。そのため、特徴点マッチングを行った後、画像間の回転角度と画像倍率の2次元ヒストグラムを作成し、正しい位置に対応されたマッチングペアに絞り込む。床指紋の照合判定としては正しく結ばれたマッチングペアが閾値以上であれば画像が照合されたと判定し、そうでなければ、不照合と判定する。

3. 特徴点抽出アルゴリズムの検証比較

3.1. 比較検証について

先行研究で使用していたAKAZEが本当に床指紋に最適な特徴点抽出アルゴリズムであるかどうかを検証する。他のOpenCVのアルゴリズムのORB、KAZEと比較を行ない、床指紋に最適な特徴点抽出アルゴリズムを確定していく。比較実験に使用する床画像は床面の材質が各階で異なる法政大学小金井キャンパス西館の1,2,4階の廊下の床画像を使用した。床画像はiPhone6Sで撮影され、本来の3024x4032の解像度を、縦横それぞれ4分の1に縮小して使用した。

3.2. アルゴリズムの比較実験

アルゴリズムの比較実験として、各階10ヶ所で床指紋の照合を行い、正しく結ばれたマッチングペア数と実行時間を比較する。各階のマッチングペア数と実行時間は10ヶ所の照合結果の平均とし、表1.1, 1.2にまとめた。AKAZEとKAZEでは、特徴点への採用を決定するthresholdを初期設定値の $1.0e-3$ と $1.0e-5$ にまで小さくした2種類で照合を行った。また、ORBでは、抽出する特徴点の最大数を5000と10000の2種類で照合を行なった。表1.1の結果から、正しい位置に対応してマッチングペアが最も多く結ばれていたのはKAZEだった。ORBの床指紋の照合にかかる実行時間がKAZE, AKAZEの1/5~1/10であり、ORBが最も速い時間で照合が行えるアルゴリズムであることがわかった。本稿では、高速化が目的であるため、ORBを最適な特徴点抽出アルゴリズムとして使用する。また、本来のORBの特徴点抽出は周辺との輝度差が著しい点を

表 1.1 マッチングペア数の比較[ペア]

アルゴリズム/画像フィルタ	1F 廊下	2F 廊下	4F 廊下
AKAZE/平均値差分 (threshold= $1.0e-3$)	234 (9.0%)	1030 (15.8%)	61 (2.4%)
AKAZE/平均値差分 (threshold= $1.0e-5$)	506 (5.8%)	2006 (17.5%)	456 (4.3%)
KAZE/グレースケール (threshold= $1.0e-3$)	499 (29.1%)	4723 (74.8%)	2329 (48.4%)
KAZE/グレースケール (threshold= $1.0e-5$)	1332 (21.1%)	6303 (71.3%)	2697 (44.3%)
ORB/グレースケール (max=5000)	158 (10.0%)	351 (18.3%)	156 (7.8%)
ORB/グレースケール (max=10000)	324 (10.5%)	782 (20.5%)	378 (9.7%)

※括弧内はマッチングペア数から正しい位置で対応して結ばれたマッチングペアの割合

表 1.2 実行時間の比較[秒]

アルゴリズム/画像フィルタ	1F 廊下	2F 廊下	4F 廊下
AKAZE/平均値差分 (threshold= $1.0e-3$)	15.9	45.9	10.4
AKAZE/平均値差分 (threshold= $1.0e-5$)	143.7	157.5	149.8
KAZE/グレースケール (threshold= $1.0e-3$)	10.7	52.4	64.2
KAZE/グレースケール (threshold= $1.0e-5$)	87.5	99.6	102.5
ORB/グレースケール (max=5000)	2.78	2.82	2.95
ORB/グレースケール (max=10000)	9.25	9.3	9.31

特徴点と捉えるため、照明などの影響で特徴点が特定の場所に偏ってしまうことがある。そのため、特徴点を画像全体に分散して抽出されるように ORB のアルゴリズムを一部修正した。

4. 床指紋の高速手法

4.1. GPGPU によるマッチングの並列処理

GPGPU を用いて特徴点マッチングを並列化することで、マッチングにかかる処理時間の短縮を行なった。CPU を使用した特徴点マッチングの平均処理時間は 7.52 秒で、GPGPU を使用した特徴点マッチングの平均処理時間は 0.12 秒だった。また、平均処理時間の結果は 10 ヶ所の床画像での照合結果の平均とした。特徴点マッチングに使用したライブラリは CPU では ORB、GPGPU では GPU に対応した CUDAORB を使用した。

4.2. マッチングペア絞り込みの高速化

先行研究ではマッチングペアが結ばれた点同士の回転角度と回転倍率を求め、ヒストグラムを作成し、正しい位置に対応されたマッチングペアに絞り込んでいた。本論文では RANSAC の手法に従い、無作為に選んだ 3 つのマッチングペアから成る三角形が 1 枚目の画像と 2 枚目の画像で相似形になっているかどうかを判断することでマッチングペアを絞り込むことにする。その結果、マッチングペアの絞り込み数は先行研究の約半分近くになった。マッチングペアの絞り込みにかかる実行時間は先行研究では平均して 0.5 秒はかかっていたのに対して、本手法では 0.01 秒で絞り込みが行えるようになった。

4.3. 床指紋のデータベース作成

マッチングのために必要な特徴点と特徴量の抽出までに掛かる実行時間は画像 1 枚あたり、約 0.7~0.8 秒ほどであった。そのため、広域化した時に特徴量抽出だけで数分の時間が必要になってしまう。そこで、事前にデータベース画像の特徴点と特徴量を抽出しバイナリーデータ化したものを、データベースに保存する。そうすることで、特徴量抽出にかかる処理時間を減少させ、データベースを読み込む時間だけで特徴点マッチングを行うことが可能になる。

4. 実験

広域実験として、法政大学小金井キャンパス西館の 4 階の廊下の画像を撮影し、画像のデータベースを作成し、撮影位置の推定実験を行った。データベースの画像は高さ約 1.5m から床に水平の状態ですべての物が写らないようにして撮影した。1 枚の画像の撮影範囲は 1.5m (縦) × 1.1m (横) であり、4 階の廊下は縦 87.5m、横 2.0m なのでデータベースの画像は 116 枚となった。本実験では、5 ヶ所の床画像で位置推定を行なった。位置推定結果は対応した場所を撮影した床画像だけ閾値以上のマッチングペアが結

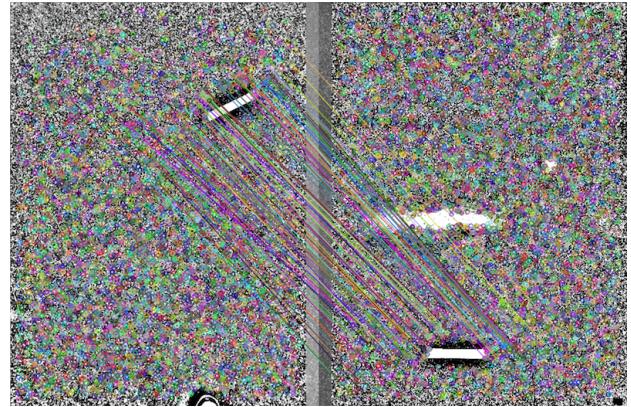


図1 マッチング結果

ばれ、位置推定に成功した。そのマッチング結果の一例を図 1 に載せる。位置推定にかかった実行時間は、先行研究の手法(AKAZE+CPU+回転角度、倍率による絞り込み)では 5078.3 秒(84 分 38.3 秒)、本手法(ORB+CPU+データベース化+相似形による絞り込み)では 933.8 秒(15 分 33.8 秒)、本手法に GPGPU を使用すると、16.5 秒だった。

5. 考察

GPGPU による並列処理によって一枚あたりの特徴点マッチング処理時間が 0.12 秒にまで減少することができた。また、特徴点、特徴量のデータベース化は一枚あたりの大幅な実行時間短縮はないが、広域化し、位置推定のために照合する床画像の枚数が増加すると、それに比例した計算量を減らすことができ、大幅な実行時間の短縮が見込めると考えている。今回、床指紋の照合にかかる実行時間は 1 秒以内に短縮できたが、実際に室内の位置推定として使用する場合、膨大な枚数の床画像と床指紋の照合を行わなくてはならない。そのため、リアルタイムで位置推定をするためには、床指紋の照合自体を並列化して、一度に何枚もの画像と床指紋の照合をする必要があると考えられる。

6. 結論

本論文では床指紋を用いた室内位置推定における実行時間の課題解決を目指した。GPGPU を用いることで特徴点マッチングの処理時間を、CPU を使用した時の 1.6% にまで短縮することができた。また、事前に特徴点、特徴量をデータベースに保存しておくことで、特徴量の計算時間も短縮することができた。また、マッチングペアの絞り込みも本手法により、約 90% 短縮することができた。また、広域実験では、87.5m の廊下で位置推定を行なった時、5078.3 秒から 16.5 秒(0.32%)までに時間を短縮することができた。

参考文献

- [1] 藤田悟, 内田薫. 床指紋を用いた位置推定. マルチメディア, 分散, 協調とモバイル (DICOMO2016) シンポジウム, 2016.