

個人ユーザのための特徴点・加速度センサを用いた 屋外 AR プラットフォームの構築

若原雅斗[†] 横井優斗[†] 鈴木正敏[‡] 濱川礼[†]

[†]中京大学情報理工学部情報システム工学科

[‡]中京大学情報理工学部機械情報工学科

1.概要

本論文では、屋外を対象とした拡張現実(以下 AR)を実現する為に必要と考えられる位置推定技術を包含したプラットフォームを提案する。開発者は本プラットフォームにより、位置情報を容易に取得可能となる。屋外 AR アプリケーション開発未経験者に提供し有効性を検証する。

2.研究背景

スマートフォン(以下端末)の普及と共に AR への関心が高まり、企業の広告等広く採用されている。今後 GlassProject などによる眼鏡型デバイスの登場によって、より身近なものになると考えられる。現在普及している AR は AR Toolkit[1] に代表されるマーカを認識する手法が多い。しかし屋外を考えたとき、マーカを配置することは現実的ではない。また GPS を用いる位置推定も行われているが、GPS には 5m 以上の誤差が存在する。マーカを利用せず位置推定を行う手法は提案されている[2,3]が個人の開発者にとって技術的にハードルが高い。図 2-1 に実装の容易さと精度の位置付けを示す。

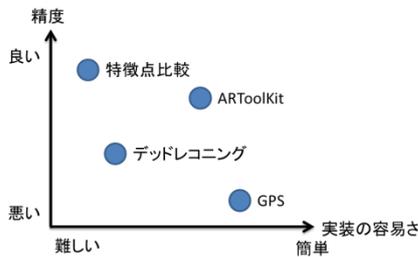


図 2-1 実装の容易さと精度の位置付け

そこで技術的要因である位置情報を取得するために、緯度経度を軸とする絶対的な位置座標を求めたのち、移動による相対的な位置座標を求める。今回、絶対位置推定に特徴点比較[2]と稜線比較、相対位置推定にデッドレコニング[3]を採用しプラットフォームを構築した

3.関連研究

AR Toolkit[1]は特殊なマーカを認識することで、位置関係を容易に取得する事が可能な、Android や iPhone 上で利用可能なライブラリである。これによりアプリケーション開発者は実際の画像処理や制御行列計算を意識することなく AR を実現する事ができる。しかし、屋外でマーカを利用するためには多くのマーカを配置する必要がある。

4.プラットフォーム

4.1. 概要

図 4.1-1 はプラットフォームの概念図である。開発者の作成するアプリケーションはプラットフォームを通して推測

Construction of an AR platform in outdoor using the Feature Point and the Acceleration Sensor for an individual user.

Masato Wakahara[†], Yuto Yokoi[†], Masatoshi Suzuki[‡], and Rei Hamakawa[†]

[†] Department of information system technology, Chukyo University.

[‡] Department of Mechano-Informatics, Chukyo University.

された位置情報を取得する事が出来る。今回、プラットフォームは名古屋駅を中心とした 1km 四方を利用可能な範囲とし、端末横持ちで操作しながら歩行することを想定した。



図 4.1-1 プラットフォーム概念図

4.2 絶対位置推定

4.2.1 処理の流れ

カメラ画像と、仮想名古屋の風景が一致する地点を探すことで位置推定を行う。全ての位置を順に探索していくと処理時間が膨大になるため、GPS の指し示す地点を探索の中心とし探索範囲を狭める。GPS には誤差範囲が定められており、範囲内に実際の地点がある。誤差範囲内を図 4.2.1-1 のように 4 分探索を行う。

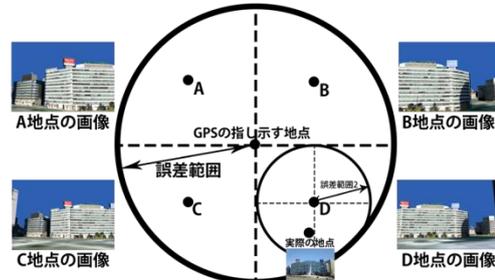


図 4.2.1-1 GPS の情報と探索範囲

端末から取得されたカメラ画像と、4 点 ABCD それぞれに対応する画像を仮想名古屋から生成し、図 4.2.1-2 のように特徴点比較と稜線比較によって相違度を求める。従来の研究[2]では特徴点のみを用いているが、都市の稜線は一意性が高く、精度の向上が可能と判断したため用いた。各相違度は大きさの基準が異なるため、式 4.2.1-3 より評価値を算出する。評価値について最も実際の地点に近いと判定された点を次の探索の中心とし探索範囲を半分に設定する。

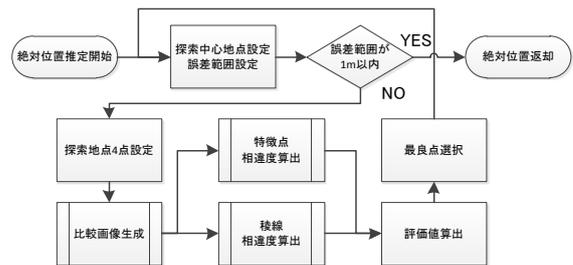


図 4.2.1-2 処理の流れ

$$\text{評価値} = \frac{\text{特徴相違度}}{\text{各地点の特徴点相違度の合計}} + \frac{\text{稜線相違度}}{\text{各地点の稜線相違度の合計}}$$

式 4.2.1-3

4.2.2 仮想名古屋

Google SketchUp[4]から建造物モデル 915 個を利用した。図 4.2.2-1 は名古屋駅周辺の仮想都市である。端末の傾きは正しいと仮定し、傾きと指定された地点に対応する画像を生成する。



図 4.2.2-1 仮想名古屋

4.2.3 特徴点比較

特徴量生成手法である SURF[5]を用いる。SURF は風景画像から特徴的な点を抽出し、画像間の輝度変化に強い特徴量を生成する。その特徴量を元に、カメラ画像と比較画像の特徴点の対応付けを行う。対応付けされた画像を図 4.2.3-1 に示す。式 4.2.3-2 を用い、特徴点間の距離の二乗の和から平均をとり相違度を算出する。



図 4.2.3-1 特徴点对応付けの例

$$R_{\text{distance}} = \frac{\sum_{i=0}^{N-1} (\text{Distance}(I_{Kp(i)}, T_{Kp(i)}))^2}{N}$$

N : 対応付けされた特徴点の数
 i : 対応付けされた特徴点
 I_{Kp} : カメラ画像の対応座標
 T_{Kp} : 比較画像の対応座標
 R_{distance} : 特徴点間距離相違度

式 4.2.3-2

4.2.4 稜線比較

図 4.2.4-1 のカメラ画像から建物と空の境界線である稜線を図 4.2.4-2 のように生成し、稜線同士の形状をテンプレートマッチング手法の一つである SAD の式 4.2.4-3 により比較し相違度を算出する。



図 4.2.4-1 カメラ画像

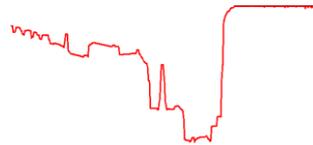


図 4.2.4-2 稜線生成画像

$$R_{SAD} = \sum_{j=0}^{N-1} \sum_{i=0}^{M-1} |I(i,j) - T(i,j)|$$

M, N : 画像の縦幅・横幅の最大値
 $I(i,j)$: カメラ画像の輝度
 $J(i,j)$: 比較画像の輝度
 R_{SAD} : 稜線形状相違度

式 4.2.4-3

4.3 相対位置推定

4.3.1 処理の流れ

端末のセンサを用い相対位置を推定するデッドレコニング[3]を用いる。センサには加速度センサと電子コンパスを用いた。加速度センサの情報を元に歩行か停止かを判定し一歩ごとに電子コンパスの指し示す方へ現在の位置から歩幅分移動をさせる。

今回、歩幅は一定であると仮定し端末所有者が入力する。

4.3.2 歩行状態判定

1 秒ごとの単位時間に含まれる鉛直方向下向きの加速度の分散を調べ、閾値を超えたとき歩行中と判断できる[3]。閾値について、大学生 11 名に実験を行い 0.21 とした。

4.3.3 歩数推定

歩行中加速度の大きさは正弦波を示す[3]。歩数を調べるには正弦波の数を数えればよい。停止中の小さな波を無視するため振幅の閾値について、大学生 11 名に実験を行い歩行中の振幅を平均 1.72 とした。

5. 評価

絶対位置推定の評価について名古屋駅周りの写真 20 枚に対して、GPS と本プラットフォームの指し示す位置座標と、正解位置との誤差を計測した。

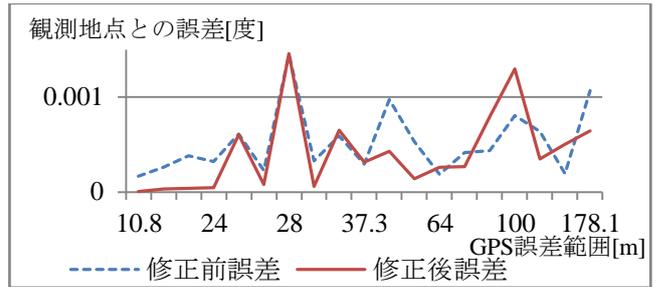


図 5-1 絶対位置誤差グラフ

図 5-1 はそのグラフである。GPS の指し示す許容誤差が 20m 以内であるとき、誤差をほぼ 0 にする事に成功した。

また、相対位置推定について大学生 11 名に対し実際に歩いた歩数と本プラットフォームの示す歩数との誤差を計測した結果 4.6% となった。更に屋外で AR のアプリケーションを制作したことの無い開発者に本プラットフォームを用い作成してもらった。図 5-2 は作成例であり、街の中に任意のメッセージを残すことが出来る。本プラットフォームを利用することで自分の作りたい物だけに集中できるという評価をもらった。



図 5-2 アプリケーション開発例

6. まとめ

AR Toolkit はマーカを用いることで技術的要因を解決し、多くの開発者は容易に開発することができた。同様に、屋外におけるプラットフォームの提供により技術的要因を解決することで個人の開発者が容易に開発を行っていくことができる。現在、絶対位置推定の精度について許容誤差が大きくなると修正前に比べ誤差が大きくなる傾向がみられた。今後多くの提案手法を実験・評価しプラットフォームに組み込むことで向上を目指す。

7. 参考文献

[1] "AR Toolkit" <http://www.hitl.washington.edu/artoolkit/>
 [2] 薄 充孝 他, "ランドマークデータベースに基づく静止画像からのカメラ位置・姿勢推定," 日本バーチャルリアリティ学会論文誌, 13(2), 2008.
 [3] 上坂 大輔 他, "手に保持されたセンサを用いた歩行者向けデッドレコニング手法の提案," 情報処理学会論文誌, 52(2), 2011
 [4] "Google Sketch Up" <http://www.sketchup.com/>
 [5] H.Bay, Tinne Tuytelaars, Luc Van Gool, "SURF: Speeded Up Robust Features", Proceedings of the 9th European Conference on Computer Vision, 2006.