

# スマートフォンでの一振り動作による 大きさの不明なARマーカまでの距離測定

勝間 亮<sup>1,a)</sup>

受付日 2016年11月10日, 再受付日 2017年3月13日,  
採録日 2017年4月18日

**概要:** 本論文では, 人間が所持したスマートフォンを一振りする動作により, スマートフォンから大きさの不明な AR マーカまでの距離を精度良く計測する方法を提案する. 単眼デジタルカメラから撮影物までの距離を計測する既存の方法としては, 異なる 2 地点から撮影した画像における撮影物の画面上の位置の差を利用するモーションステレオ法がある. しかし, 人間が振るスマートフォンの移動距離は一定ではないため, スマートフォンに搭載された加速度センサと方位センサを用いて移動距離を測定する必要があるが, 各センサの計測値には誤差が必ず含まれる. そこで, カメラ移動の縦成分もしくは横成分のどちらか一方が分かればカメラとマーカとの距離の測定が可能であることに着目し, 両方の成分それぞれについてカメラとマーカとの距離を測定する. 縦成分と横成分の 2 種のアプローチから測定されたカメラとマーカとの距離の加重平均をとることで誤差の大きい成分を排除し, 推定精度を向上させる. 実験の結果, 4[m] の距離推定実験において, スマートフォンを横移動および縦移動した場合, 提案手法の方が従来のモーションステレオ法よりも平均的に約 8% の誤差軽減を達成し, 極端な精度の悪化を軽減できたことを確認した.

キーワード: モーションステレオ, 距離測定, スマートフォン, 加速度センサ

## Distance Estimation between Smartphone and AR Marker without Size Information by One Shaking Behavior

RYO KATSUMA<sup>1,a)</sup>

Received: November 10, 2016, Revised: March 13, 2017,  
Accepted: April 18, 2017

**Abstract:** In this paper, we propose a method for measuring shooting distance between a camera on a smartphone and an AR marker with high accuracy by one shaking the smartphone. There is an existing method for estimating shooting distance with single camera called Motion Stereo that calculates by the difference of the onscreen object positions taken at different two points. The accuracy of the shooting distance depends on the accuracy of the distance between the two shooting points. In Motion Stereo by the camera on the smartphone, it is necessary to use the accelerometer for measuring the moving distance. However, the accelerometer always has error range. Then, we propose a new Motion Stereo that estimates the shooting distance by two kinds of ways; vertical and horizontal component of the camera motion. In order to improve the estimation accuracy by removing a component including a large error, the proposed method calculates the weighted average of two kinds of shooting distances. The experimental results show that the proposed method reduces estimation errors 8% less than existing Motion Stereo and avoids extreme dropping of the accuracy.

**Keywords:** motion stereo, distance estimation, smartphones, accelerometer

<sup>1</sup> 大阪府立大学  
Osaka Prefecture University, Sakai, Osaka 599-8531, Japan  
<sup>a)</sup> katsuma@cs.osakafu-u.ac.jp

### 1. はじめに

近年, スマートフォンなどで撮影した現実世界の映像に

付加情報を表示する AR (Augmented Reality) 技術が研究されている。AR では画面に表示される情報の位置や角度が現実世界と合致している必要があるため、カメラの位置や姿勢を検出しなければならない。そこで、AR マーカと呼ばれるパターン画像をあらかじめ設置しておき、撮影された映像からマーカを自動的に読み取ることで、カメラの位置や姿勢を特定する方法が提案されてきている [1]。一般的に、AR マーカは正方形であり、その四隅もしくは外周は識別しやすくするための模様が大きく縁取られ、内部の模様で固有に割り振られた ID を示している。マーカを遠くから読み取る場合は大きいマーカでないと判別ができず、逆に近くから読み取る場合は小さいマーカでないと画面に収まりきらないため、マーカの大きさは用途によって変更する必要があり、統一することは難しい。このような AR マーカに対して、スマートフォンのカメラでマーカの ID 情報を読み取り、ネットワークを介してマーカの大きさや位置情報、その周辺に関する様々な情報を取得して利用者に提示する AR ナビゲーションサービスが考えられている [2], [3]。正しい情報提示のためにはカメラの位置情報が必要であり、GPS が使用できない環境下で位置情報を得るために必要な情報の 1 つとして、カメラとマーカ間の距離情報があげられる。ネットワーク上からマーカの大きさの情報を取得できる場合、実際のマーカの大きさと画面に映っているマーカの大きさの比率からカメラとマーカ間の距離を測定できる [4]。その結果、マーカの位置情報をダウンロードすることでカメラの位置を特定することができる。

しかし、屋内で電波を使用すると、壁や天井による電波の乱反射が発生し、他の機器から発信された電波と干渉を起こす場合があり、屋内ではできるだけ電波を飛ばさない位置推定手法が求められる。それを実現するためには、AR マーカに ID と位置推定に必要な情報を盛り込み、ネットワークを使用せずカメラのみを用いてユーザの位置推定を行う必要がある。AR マーカに情報を持たせるために、マーカを白黒のビットパターンを表す模様にする研究が行われているが、マーカに盛り込むことができるデータ量が少ないことが問題となっている [5]。

そこで本研究では、マーカの所持するデータ量を減らすため、マーカの大きさを表す情報がない状態でカメラとマーカ間の距離を測定する手法を提案する。提案手法では、人間がカメラと加速度センサ、ジャイロセンサ、磁気センサが一体化したスマートフォンを持ち、カメラでマーカを撮影しながらスマートフォンを 1 度移動させるだけでモーションステレオ法によりマーカまでの距離を測定する。モーションステレオ法とは、1 台のカメラを移動させながら異なる 2 地点から同一の物体を撮影し、画面に映る物体の位置のずれから物体までの距離を測定する方法である。このとき、2 地点間の距離とカメラの姿勢差が判明し

ている必要があり、加速度センサ、ジャイロセンサ、磁気センサによりそれらを計測する。モーションステレオを用いた従来手法では、奥行き成分を除いた移動距離からカメラと対象物までの距離測定を行っている。しかし、加速度センサ、ジャイロセンサ、磁気センサの誤差が引き起こすカメラの移動距離の誤差により、カメラとマーカ間の距離測定精度が低下するという問題がある。ここで、本論文では、誤差とは推定した値と実際の値の差分量のことをいう。そこで、提案手法では移動方向の成分分解を行うことで、誤差の大きい成分を排除して距離測定を行う。カメラとマーカ間の距離の測定は、カメラ移動の縦成分もしくは横成分のどちらか一方が分かれば可能であり、提案手法では両方の成分それぞれについてカメラとマーカ間の距離を測定する。そして、縦成分と横成分の 2 種のアプローチから測定されたカメラとマーカ間の距離の加重平均をとることで精度を向上させる。

予備実験では、あらかじめスマートフォンを持った手の一振りの動き方とスマートフォンの移動距離測定精度の関係性を調査した。その結果、移動距離が小さいほど移動距離に対する誤差の大きさが増大する性質があることが予測できた。その性質をもとにして、画面上のマーカの移動方向から推測可能なスマートフォンの移動方向に応じて決定される加重平均の関数を設定した。また、センサ値の誤差を軽減するためのローパスフィルタの係数も予備実験により決定した。実験の結果、4 [m] の距離推定実験において、スマートフォンを横移動および縦移動した場合、提案手法の方が従来手法よりも平均的に約 8% の誤差軽減を達成し、極端な精度の悪化を軽減できたことを確認した。また、スマートフォンを斜め 45 度方向へ移動させることで、距離推定精度が良くなることが分かった。

以降、2 章では関連研究、3 章では問題設定について述べる。4 章で解決方法を述べ、5 章で提案手法の有効性の評価について述べる。

## 2. 関連研究

### AR マーカ

正方形の AR マーカの四方や四隅に特徴的な模様を挿入することによってカメラによる認識をしやすくしており、ARToolKit などでは高精度で自動認識可能である [6]。現在普及している QR コードと似ており、注意深く接写することが前提の QR コードは最大で 3 万ビットの情報を格納できる。しかし、柴田らが提案したマーカ内部の白黒のビット表示によりデータを格納できる AR マーカでは、ある程度遠方から映すことが前提であるため大きさに制限があり、約 25 [bit] 程度の情報量を格納できるととどまっている [5]。そのため、AR マーカに格納する情報量の削減は非常に重要な課題である。著者の知る限り、本論文は情報の削減対象としてマーカの大きさをあげた初めての論文で

ある。

### 特徴点までの距離推定

電波を用いない位置推定方式として、カメラを使用して自身や対象物の位置推定を行う研究は数多くあり、その中でも目印として設置したマーカとカメラとの距離を測定するアプリケーションには QPToolKit などが存在する [4], [7]. しかし、あらかじめマーカの大きさを知っておく必要があり、マーカの大きさ情報なしで距離測定を行うことはできない。それに対して、モーシヨンステレオ法と呼ばれる単眼カメラで特徴点までの距離を測定する方式が提案されてきている [8], [9]. モーシヨンステレオ法では、カメラの移動距離と撮影された特徴点の移動距離の比率から、カメラと特徴点までの距離を推定している。この方式はカメラの移動距離が判明していれば距離推定が可能であり、モータにより移動量や方向を精密に制御可能なロボットが様々な物体までの距離を認識し、マップなどを生成する際によく用いられる [11]. しかし、たとえばスマートフォンの加速度センサの誤差などによる移動距離の計測誤差があると、特徴点までの距離推定精度が悪化してしまうという問題がある。特に人が手に持つスマートフォンはつねにぶれがあり、加速度センサの値も誤差がつねに発生している状態なため [12], これらの従来のモーシヨンステレオ方式では距離推定精度が悪くなる可能性がある。そのため、誤差を軽減する方式が求められている。

これらを考慮すると、以下の要件を満たす新たな距離推定手法が必要である。

- 電波を使用しない単眼カメラによるモーシヨンステレオ法
- 一般に普及しているスマートフォンの使用
- AR マーカの大きさ情報の不使用
- 加速度センサの誤差の影響の軽減

よって、本論文では上記を満たす新たなモーシヨンステレオ法を提案する。

### 3. 問題設定

本章ではカメラに付属した加速度センサとジャイロセンサを用いて対象物までの距離をモーシヨンステレオで測定するための問題設定を述べる。

#### 環境設定

絶対座標系 (4.1 節で説明) に点  $P$  と、焦点距離  $f$  の標準レンズを搭載したカメラ  $C$  がある。カメラで撮影しているときの画面上に映った点  $P$  を点  $Q$  と呼び、点  $Q$  の画面上の座標を  $(q_x, q_y)$  とする。絶対座標系はカメラの運動履歴によって軸が変化することはない。絶対座標系のほかに、図 1 で示すカメラ座標系を設定する。カメラ座標系はカメラを原点とした座標系であり、カメラの運動によって

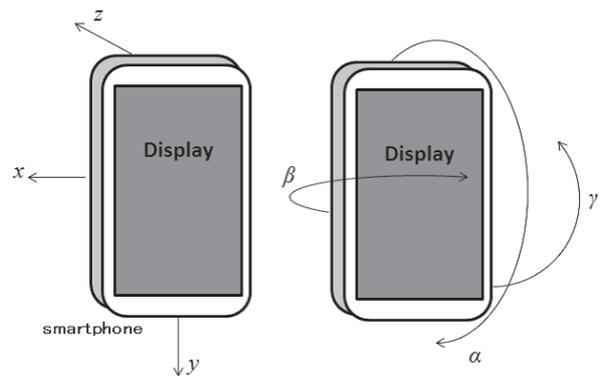


図 1 カメラ座標系の軸設定

Fig. 1 Axes.

絶対座標系に対して軸が回転および移動していく。また、今後は、 $x$  座標の値が増え、かつ  $y$  座標と  $z$  座標が変化しない移動方向のことを  $x$  軸方向と呼び、 $y$  軸方向や  $z$  軸方向も同様とする。

#### 一振り動作

ユーザはカメラを手に持ちながら距離を計測したい点  $P$  にカメラを向けて静止し、その後に歩行することなく点  $P$  を映しつつカメラを移動させてから再び静止する。このとき、激しい不規則な動きや振動は起こらないものとする。このユーザの動作のみから、カメラ  $C$  から点  $P$  までの距離  $d$  を求めるために、カメラがどのように動いたかを知る必要がある。そこで、加速度と方角の情報を利用する。

#### 各種センサ

カメラには三軸加速度センサとジャイロセンサと磁気センサが搭載されているとする。これらのセンサを搭載したカメラは片手で持ち運び可能であるとする。このような製品の代表例として、スマートフォンがあげられる。加速度センサはカメラ座標系での  $(x, y, z)$  成分の加速度を検出することができる。ジャイロセンサと磁気センサを組み合わせることで、デバイスの姿勢を絶対座標系で取得できるため、これらをまとめて方角センサと呼ぶ。方角センサは絶対座標系での  $xyz$  軸に対応するヨーピッチロール  $(\alpha, \beta, \gamma)$  で表される姿勢を検出することができる。これらのセンサの各成分は図 1 の矢印方向を正の値、その逆方向を負の値で表されるとする。加速度センサと方角センサはつねに同時刻にサンプリングできるものとし、サンプリングの周期  $I$  は十分に短いものとする。カメラが移動開始した時刻を  $t_0$  としたとき、 $n$  回目のサンプリング時刻  $t_0 + nI$  で検出された加速度成分の値は  $(x_n, y_n, z_n)$ 、方角成分の値は  $(\alpha_n, \beta_n, \gamma_n)$  で表す。

#### 対象問題と制約条件

本問題は、カメラで点  $P$  を撮影しながら移動し、カメラ

に付属した三軸加速度センサと方角センサの各時刻での値と画面上の点  $Q$  の座標の変化から、移動後のカメラ  $C$  と点  $P$  との距離  $d$  をできるだけ正確に求めることである。ここで、問題を解く際に大きな支障が出ないように次の制約条件を設ける。

- (1) カメラが移動開始した時刻  $t_0$  から移動終了した時刻  $t_{end}$  まで各種センサとカメラは稼働し続ける。
- (2) 時刻  $t_0$  ではカメラは点  $P$  に対して静止しているものとする。
- (3) 時刻  $t_0$  および時刻  $t_{end}$  では点  $Q$  が画面内に存在するものとする。

(1) に関して、運動途中のセンサの情報や運動後の画像が取得できなくなった場合、カメラの動きを知ることができなくなるため、 $d$  が測定不能となる。(2) に関して、時刻  $t_0$  でカメラが点  $P$  に対して等速直線運動をしている場合は加速度センサや方角センサで検出できないうえに、運動の方向によって点  $Q$  の座標  $(q_x, q_y)$  は様々な方向に移動してしまうため、カメラの速度に応じた大きさのノイズが点  $Q$  の座標の変化に加わり、 $d$  の測定精度が著しく低下する。(3) に関して、時刻  $t_0$  から時刻  $t_{end}$  までの間の点  $Q$  の座標の変化度合いが計測できなければ、 $d$  が測定不能となる。

#### 4. 提案手法

本章では、3章で述べた問題を解決するための方法について述べる。提案手法では、画面上の点  $Q$  の  $x$  座標  $q_x$  の変化に関わる絶対座標系の移動成分と方角成分のみからカメラ  $C$  から点  $P$  までの距離  $d$  を算出し、 $Q$  の  $y$  座標  $q_y$  の変化に関わる絶対座標系の移動成分と方角成分のみから  $d$  をもう一度算出する。そして、それらの加重平均をとることで測定誤差を軽減する。

##### 4.1 カメラ座標系から絶対座標系への変換

3章で述べたとおり、カメラは一連の移動を行った後に停止する時刻  $t_{end}$  から静止状態に入る。このときのカメラの姿勢  $(\alpha_{end}, \beta_{end}, \gamma_{end})$  を以降で絶対座標系とする。

加速度センサはカメラ座標系の各成分の値  $(x, y, z)$  を検出するため、これらの値を絶対座標系の加速度に変換する必要がある。時刻  $t_0 + nI$  で、方角センサが  $(\alpha_n, \beta_n, \gamma_n)$ 、加速度センサが  $(x_n, y_n, z_n)$  を計測したとする。このとき、絶対座標系のベクトルをカメラ座標系に変換するためのロドリゲスの回転行列  $R_n$  は、 $\phi_n = \alpha_n - \alpha_{end}$ 、 $\theta_n = \beta_n - \beta_{end}$ 、 $\psi_n = \gamma_n - \gamma_{end}$  とおいたとき、ページ下に示す式 (1) となる。

$$R_n = \begin{pmatrix} \cos(\theta_n)\cos(\phi_n) & \sin(\psi_n)\sin(\theta_n)\cos(\phi_n) - \cos(\psi_n)\sin(\phi_n) & \sin(\psi_n)\sin(\phi_n) + \cos(\psi_n)\sin(\theta_n)\cos(\phi_n) \\ \cos(\theta_n)\sin(\phi_n) & \sin(\psi_n)\sin(\theta_n)\sin(\phi_n) - \cos(\psi_n)\cos(\phi_n) & -\sin(\psi_n)\cos(\phi_n) + \cos(\psi_n)\sin(\theta_n)\sin(\phi_n) \\ -\sin(\theta_n) & \sin(\psi_n)\cos(\theta_n) & \cos(\psi_n)\cos(\theta_n) \end{pmatrix} \quad (1)$$

$R_n$  を用いると、絶対座標系の加速度  $(x'_n, y'_n, z'_n)$  とカメラ座標系の各成分  $(x_n, y_n, z_n)$  の関係は式 (2) となる。

$$R_n \begin{pmatrix} x'_n \\ y'_n \\ z'_n \end{pmatrix} = \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} \quad (2)$$

ここで、 $R_n$  は直行行列であることが知られており、 $R_n^{-1} = R_n^T$  であるため、最終的にカメラ座標系で得られた加速度情報を絶対座標系での加速度情報に変換するための式 (3) が得られる。

$$\begin{pmatrix} x'_n \\ y'_n \\ z'_n \end{pmatrix} = R_n^T \begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} \quad (3)$$

今後、特に断りのない限り  $x$  軸、 $y$  軸、 $z$  軸と書くと絶対座標系のそれらを指すものとする。

##### 4.2 画面上の $x$ 軸方向に関わる動作

画面上の点  $Q$  の  $x$  座標  $q_x$  の変化に関わる動作は、次のものがあげられる。

- $x$  軸方向へのカメラの移動
- $z$  軸方向へのカメラの移動
- $y$  軸を中心としたカメラの回転運動
- $z$  軸を中心としたカメラの回転運動

これらの動作が  $q_x$  に与える変化量について、以下で述べる。

##### $x$ 軸方向へのカメラの移動

カメラ  $C$  の仮想スクリーンと点  $P$  の間の距離  $d_v$  と、カメラの焦点距離  $f$ 、カメラの  $x$  軸方向への移動距離  $x$  を用いて、移動後の画面上の点  $Q$  の  $x$  座標  $MoveX(q_x, d_v, f, x)$  は以下の式 (4) で表される。

$$MoveX(q_x, d_v, f, x) = q_x + x \frac{f}{d_v + f} \quad (4)$$

ここで、 $d_v$  をカメラ  $C$  と点  $P$  の間の距離  $d$  で表すと、 $d_v = \frac{d}{\sqrt{1+(\frac{q_x}{f})^2}} - f$  であるため、式 (4) は以下の式 (5) に変形できる。

$$MoveX\left(q_x, \frac{d}{\sqrt{1+(\frac{q_x}{f})^2}} - f, f, x\right) = q_x + x \frac{f}{\frac{d}{\sqrt{1+(\frac{q_x}{f})^2}}} \quad (5)$$

##### $z$ 軸方向へのカメラの移動

移動前のカメラ  $C$  の仮想スクリーンと点  $P$  の間の距

離  $d_v$  と、カメラの焦点距離  $f$ 、カメラの  $z$  軸方向への移動距離  $z$  を用いて、移動後の画面上の点  $Q$  の  $x$  座標  $MoveZ(q_x, d_v, f, z)$  は以下の式 (6) で表される。

$$MoveZ(q_x, d_v, f, z) = q_x \left( 1 + \frac{z}{d_v + f - z} \right) \quad (6)$$

ここで、 $d_v$  の代わりに  $d$  で式 (6) 表すと、以下の式 (7) となる。

$$\begin{aligned} MoveZ(q_x, \frac{d}{\sqrt{1 + (\frac{q_x}{f})^2}} - f, f, z) \\ = q_x \left( 1 + \frac{z}{\frac{d}{\sqrt{1 + (\frac{q_x}{f})^2}} - z} \right) \end{aligned} \quad (7)$$

#### y 軸を中心としたカメラの回転運動

カメラ  $C$  と点  $P$  の間の距離  $d$  と、カメラの焦点距離  $f$ 、カメラ座標系の  $y$  軸を中心とした回転量  $\beta$  [rad] を用いて、回転後の画面上の点  $Q$  の  $x$  座標  $Rolly(q_x, f, \beta)$  は以下の式 (8) で表される。

$$Rolly(q_x, f, \beta) = \frac{q_x + f \tan(\beta)}{1 - \frac{q_x}{f} \tan(\beta)} \quad (8)$$

#### z 軸を中心としたカメラの回転運動

カメラ座標系の  $z$  軸を中心とした回転量  $\gamma$  [rad] を用いて、回転後の画面上の点  $Q$  の  $x$  座標  $RollZ(q_x, q_y, \gamma)$  は以下の式 (9) で表される。

$$RollZ(q_x, q_y, \gamma) = q_x \cos(\gamma) - q_y \sin(\gamma) \quad (9)$$

#### 4.2.1 移動距離の計測

各センサのサンプリング周期を  $I$  としたとき、時刻  $t_0 + nI$  から  $t_0 + (n + 1)I$  の間の加速度は不明であるため、何らかの補間を行う必要がある。様々な補間の方法が存在するが、ここでは台形補間法を用いる。その理由として、本研究で想定しているユーザによるカメラの動かし方がきわめて単純であるため、補間が必要な範囲内での極端な加速度の乱高下が起きにくいことと、データ数が増加しても精度が悪くならないことがあげられる。時刻  $t_0 + nI$  から  $t_0 + (n + 1)I$  の間に  $x$  軸方向に移動した距離  $h_{x(n,n+1)}$  は、以下の式 (10) で表される。

$$h_{x(n,n+1)} = \frac{(s_x(n) + s_x(n+1))I}{2} \quad (10)$$

ここで、 $s_x(k)$  は時刻  $t_0 + kI$  でのカメラの  $x$  軸方向の速度を表し、最初は静止しているため  $k = 0$  のとき  $s_x(0) = 0$  である。それ以外の場合の  $s_x(k)$  は式 (11) で記述できる。

$$s_x(k) = \sum_{j=0}^{k-1} s_x(j) + \frac{(x'_{k-1} + x'_k)I}{2} \quad (11)$$

ただし、 $x'_k$  は時刻  $t_0 + kI$  における  $x$  軸方向の加速度である。  $y$  軸方向および  $z$  軸方向の移動距離も、 $y'_i$  および  $z'_i$  をそれぞれ用いて同様の式で記述できるため、省略する。

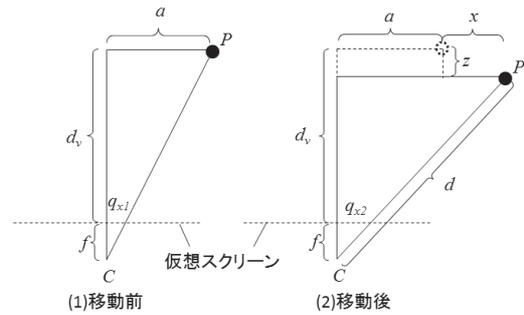


図 2 図形による表現

Fig. 2 Expression by figures.

#### 4.2.2 x 軸成分からの距離推定

$x$  軸方向に関わる運動は式 (4), (6), (8), (9) で表されるが、このうち求めるべきカメラ  $C$  と点  $P$  の距離  $d$  を含むのは式 (4), (6) である。つまり、回転運動による点  $Q$  の  $x$  座標のずれ幅は距離にかかわらないため、本問題ではノイズとなる。本手法では、回転による  $x$  座標のずれを除去してから、移動量と画面上の点  $Q$  の変化量より  $d$  を求める。

カメラ移動前の画面上の点  $Q$  の座標を  $(q_{x0}, q_{y0})$  とする。カメラの移動後の姿勢を  $(\alpha_{t_{end}}, \beta_{t_{end}}, \gamma_{t_{end}})$  とする。回転による  $x$  座標のずれを除去したカメラ移動前の点  $Q$  の  $x$  座標  $q_{x1}$  は式 (8) と式 (9) をもとにして、以下の式 (12) で表される。

$$q_{x1} = Rolly(RollZ(q_{x0}, q_{y0}, \gamma_0 - \gamma_{t_{end}}), f, \beta_0 - \beta_{t_{end}}) \quad (12)$$

回転の影響を除去した状態で、カメラの仮想スクリーンと点  $P$  との距離を  $d_v$  とすると、カメラ  $C$  と点  $P$  の  $z$  軸方向の距離は  $d_v + f$  となる。  $C$  と  $P$  の  $x$  軸方向の距離を  $a$  とおく。これを図 2(1) で示す。  $C$  の直近に存在する点線はカメラの仮想スクリーンである。ここから、 $x$  軸方向に  $x$ 、 $z$  軸方向に  $z$  だけ移動をした場合、図 2(2) の状態になる。  $x$  は  $x$  軸方向の移動距離、 $q_{x2}$  は式 (12) と同様に移動後の画面上の点  $Q$  の  $x$  座標から回転の影響を除去したものである。このときの  $d$  が本問題で求めたい値であり、以下で  $d$  を求めるための詳細を説明する。

いま、 $f$ 、 $q_{x1}$ 、 $q_{x2}$ 、 $x$ 、 $z$  は既知であり、 $a$  と  $d_v$  と  $d$  が未知である。図 2(1), (2) から、図形の性質より、以下の連立方程式 (13) が得られる。

$$\begin{cases} \frac{a}{d_v + f} = \frac{q_{x1}}{f} \\ \frac{a + x}{d_v + f - z} = \frac{q_{x2}}{f} \\ d = \frac{(d_v + f - z) \sqrt{q_{x2}^2 + f^2}}{f} \end{cases} \quad (13)$$

これを  $d$  について解くと、式 (14) となる。

$$d = \frac{fx + q_{x1}z \sqrt{q_{x2}^2 + f^2}}{f(q_{x2} - q_{x1})} \quad (14)$$

ただし、 $q_{x2} = q_{x1}$  の場合は  $d$  を求めることができない。

### 4.3 画面上の y 軸方向に関わる動作

本節は、4.3 節で説明した  $q_x$  の変化に関わる動作と同様で、縦横の関係を入れ替えたものであり、必要な数式以外は省略して記述する。

画面上の点  $Q$  の  $y$  座標  $q_y$  の変化に関わる動作は、次のものがあげられる。

- $y$  軸方向へのカメラの移動
- $z$  軸方向へのカメラの移動
- $x$  軸を中心としたカメラの回転運動
- $z$  軸を中心としたカメラの回転運動

これらの動作が  $q_y$  に与える変化量について、以下で述べる。

#### y 軸方向へのカメラの移動

カメラ  $C$  と点  $P$  の間の距離  $d$ 、カメラの焦点距離  $f$ 、カメラの  $x$  軸方向への移動距離  $y$  を用いて、移動後の画面上の点  $Q$  の  $y$  座標  $MoveY(q_x, d, f, y)$  は以下の式 (15) で表される。

$$MoveY(q_y, \frac{d}{\sqrt{1+(\frac{q_y}{f})^2}} - f, f, y) = q_y + y \frac{f}{\sqrt{1+(\frac{q_y}{f})^2}} \quad (15)$$

#### x 軸を中心としたカメラの回転運動

カメラ  $C$  と点  $P$  の間の距離  $d$  と、カメラの焦点距離  $f$ 、カメラの  $x$  軸を中心とした回転量  $\alpha$  [rad] を用いて、回転後の画面上の点  $Q$  の  $x$  座標  $RollX(q_y, f, \alpha)$  は以下の式 (16) で表される。

$$RollX(q_y, f, \alpha) = \frac{q_y + f \tan(\alpha)}{1 - \frac{q_y}{f} \tan(\alpha)} \quad (16)$$

#### 4.3.1 y 軸成分からの距離推定

回転による  $y$  座標のずれを除去したカメラ移動前の点  $Q$  の  $y$  座標  $q_{y1}$  は式 (16) と式 (9) をもとにして、以下の式 (17) で表される。

$$q_{y1} = RollX(RollZ(q_{x0}, q_{y0}, \gamma_0 - \gamma_{t_{end}}), f, \alpha_0 - \alpha_{t_{end}}) \quad (17)$$

回転による  $y$  座標のずれの除去後、カメラと点  $P$  の距離  $d$  は式 (18) で求めることができる。

$$d = \frac{fy + q_{y1}z\sqrt{q_{y2}^2 + f^2}}{f(q_{y2} - q_{y1})} \quad (18)$$

ただし、 $y$  は  $y$  軸方向の移動距離、 $q_{y2}$  は移動後の画面上の点  $Q$  の  $y$  座標から回転の影響を除去したものである。

### 4.4 加重平均

カメラと点  $P$  までの距離  $d$  について、 $x$  軸方向の成分から算出したものを  $d_x$ 、 $y$  軸方向の成分から算出したものを  $d_y$  とする。もしこれらの値が同等の誤差を含む値であれば

大数の法則により、 $d_x$  と  $d_y$  の平均値  $\frac{d_x+d_y}{2}$  を最終的な距離推定結果とすることで、 $d_x$  もしくは  $d_y$  を最終的な距離推定結果とするよりも誤差を軽減できる。しかし、たとえば  $d_y$  の誤差が  $d_y$  の値に比べて非常に大きい場合、 $\frac{d_x+d_y}{2}$  よりも  $d_x$  のみを最終的な距離推定結果とした方が精度が良くなる。そこで、 $d_x$  および  $d_y$  における、 $d_x$  および  $d_y$  の誤差の占める割合  $\rho_x, \rho_y$  を引数とした重み関数  $w(\rho_x, \rho_y)$  を用いて、最終的な距離推定結果を式 (19) とする。これを以降、加重平均関数と呼ぶ。

$$w(\rho_x, \rho_y)d_x + (1 - w(\rho_x, \rho_y))d_y \quad (19)$$

ただし、 $w(\rho_x, \rho_y)$  の値域は 0 以上 1 以下とし、センサの誤差特性、スマートフォンを持つ人の癖などで最適な関数は変化すると考えられるため、使用場面に応じて適切に設定する必要がある。

## 5. 評価実験

本章では、提案手法の性能を評価するために、AR マーカ、スマートフォン、Web カメラ、PC を用意して行った実験について述べる。まず、人がスマートフォンを持って一振りする動作について、 $x$  軸成分および  $y$  軸成分の移動距離と加速度センサの誤差の関係性を調査する予備実験を行った。その結果をもとに重み関数  $w$  を決定し、提案手法による距離推定の精度と、従来から用いられる  $xy$  座標平面での移動距離を用いたモーションステレオ法との比較により、提案手法の性能を評価した。使用した機器とアプリケーションを表 1 に示す。

### 5.1 予備実験

本節では 3 種類の予備実験について述べる。まず人が手に持ったスマートフォンで撮影対象物を映しながら移動させたときの、回転成分の大きさを計測した。次にセンサの取得した値の誤差を軽減するためのローパスフィルタの係数を調整するための実験と、人が手に持ったスマートフォンを移動させたときの計測誤差の大きさを測定する実験を行った。

#### 5.1.1 回転成分の検出

本提案手法の説明をした 2 名 A, B と前提知識のない 2

表 1 使用した機器

Table 1 Devices.

|                |   |
|----------------|---|
| カメラ            | Logicool HD Webcam c525 (解像度 1,280 × 720) |
| スマートフォン        | SONY Xperia SO-04H                        |
| 加速度センサ, 方角センサ  | SONY Xperia SO-04H 内蔵の各種センサ               |
| センサ値取得アプリケーション | Sensor Kinetics [13]                      |
| AR マーカ         | 210.0 mm × 210.0 mm ビットパターン画像             |

表 2 スマートフォン移動の際の回転量 [rad]

Table 2 Amount of rotation by moving smartphone [rad].

| 被験者 | $\alpha$ 成分 | $\beta$ 成分 | $\gamma$ 成分 |
|-----|-------------|------------|-------------|
| A   | 0.08        | 0.14       | 0.05        |
| B   | 0.07        | 0.13       | 0.09        |
| C   | 0.12        | 0.14       | 0.07        |
| D   | 0.09        | 0.17       | 0.05        |

名 C, D の合計 4 名で、人が手に持ったスマートフォンで撮影対象物を映しながら移動させたときの回転成分の大きさを計測する実験を行った。1 名につき 5 回ずつスマートフォンを移動させてもらい、移動前と移動後の方角センサの差の平均値を算出した。その結果を表 2 に示す。いずれの被験者も、目立った回転量はなかった。このことから、カメラ座標系と絶対座標系は大きくは異ならないため、三軸加速度センサのある軸に起こる累積誤差は、絶対座標系の該当軸においても同程度の誤差であることがいえる。

5.1.2 ローパスフィルタの係数の決定

加速度センサの取得した値の誤差をできるだけ軽減させるためには、ローパスフィルタ [10] を用いてセンサ取得値を補正する必要がある。ローパスフィルタは計測値の高周波成分を除去するためのもので、誤差の原因となる細かなぶれを軽減することができる。ローパスフィルタを用いると、加速度センサ  $v$  における時刻  $t_0 + (n + 1)I$  の加速度は、時刻  $t_0 + (n + 1)I$  の計測値  $v_{n+1}$  と時刻  $t_0 + nI$  の計測値  $v_n$  を用いて式 (20) で表される。

$$bv_{n+1} + (1 - b)v_n \tag{20}$$

ここで、 $b$  は 0 以上 1 未満の係数であり、計測値の精度を上げるためには適切に設定しなければならない。

人がスマートフォンを持って移動させたときの適切なローパスフィルタ係数  $b$  を求める実験を行った。移動させる距離は 10 cm から 50 cm まで、10 cm 刻みで変更した。あらかじめ移動開始地点と移動終了地点を決めておき、できるだけ x 軸方向のみの移動成分になるように、スマートフォンの向きを考慮しつつ移動させた。得られたセンサデータにローパスフィルタをかけた値から式 (11) および式 (10) を用いて推定した移動距離について、できるだけ実際の移動距離に近づくように、小数点以下第 5 位まで調整した係数  $b$  の準最適解を求めた。係数  $b$  の最適値は試行ごとに変化するため、各距離につき 20 回試行して係数  $b$  の平均値を求めた。その結果を表 3 に示す。

5.1.3 移動誤差の計測

ここでは、移動距離の長さとの誤差の関係性について調査するための予備実験について述べる。

スマートフォンの動かし方を 5.1.2 項で述べた内容と同様にして、移動させる距離を 10 cm から 50 cm まで 10 cm 刻みで変更しつつ加速度センサの値を取得した。その値に対し、今度は移動距離に応じて係数を表 3 を用いて設定し

表 3 ローパスフィルタ係数  $b$  の準最適解

Table 3 semi-optimal value of  $b$ .

| 移動距離  | 準最適な係数 $b$ の平均値 |
|-------|-----------------|
| 10 cm | 0.05638         |
| 20 cm | 0.10481         |
| 30 cm | 0.16694         |
| 40 cm | 0.13216         |
| 50 cm | 0.15742         |

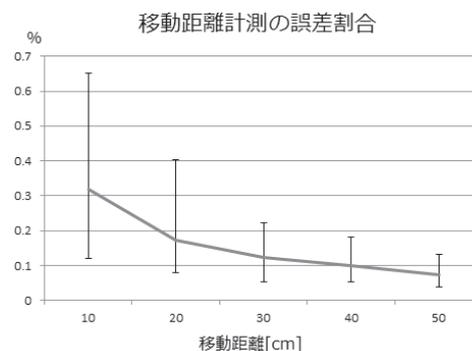


図 3 移動距離計測の誤差

Fig. 3 Error of moving distance.

たローパスフィルタをかけた。そして、式 (11) および式 (10) を用いて推定した移動距離と実際の移動距離との誤差を求めた。各距離につき 10 回試行して誤差の絶対値の平均値を求め、実際に移動した距離  $d_{real}$  に対する測定された移動距離  $d_{estm}$  の誤差の占める割合  $100 * \frac{|d_{real} - d_{estm}|}{d_{real}}$  を求めた。以降、これを移動距離測定の誤差割合と呼ぶ。その結果を図 3 に示す。

図 3 から、移動距離が小さくなればなるほど誤差割合が大きくなることが分かった。特に、移動距離 10 cm のときに誤差割合が 32% 程度と非常に大きな値となっている。

5.1.4 加重平均関数の設定

図 3 の結果から、以降の実験において、手に持ったスマートフォンを移動させる動きをするとき、距離の増加とともに誤差割合が減少すると仮定する。移動距離を x 軸成分  $d_x$  と y 軸成分  $d_y$  に分解すると、5.1.1 項で述べた結果より、 $d_x$  と  $d_y$  の関係は画面上の点  $Q$  から回転成分を除去した x 座標  $q_{x2}$  および y 座標  $q_{y2}$  を用いて  $\frac{d_y}{d_x} \approx \frac{q_{y2}}{q_{x2}}$  と近似できる。これにより、提案手法における加重平均関数  $w(\rho_x, \rho_y)$  は、 $w(q_{x2}, q_{y2})$  と表すことができるため、以降の実験では式 (21) を用いることとした。

$$w(q_{x2}, q_{y2}) = \begin{cases} 1 & (\frac{q_{y2}}{q_{x2}} < \frac{1}{\sqrt{3}}) \\ \frac{q_{x2}}{q_{x2} + q_{y2}} & (\frac{1}{\sqrt{3}} \leq \frac{q_{y2}}{q_{x2}} \leq \sqrt{3}) \\ 0 & (\sqrt{3} < \frac{q_{y2}}{q_{x2}}) \end{cases} \tag{21}$$

ただし、ここでの場合分けの閾値  $\sqrt{3}$  および  $\frac{1}{\sqrt{3}}$  は予備実験を繰り返したことによる経験則に基づいて決定しており、最適な閾値を示しているものではない。このような閾

表 4 本実験でのローパスフィルタ係数  $b$  の設定  
Table 4 Settings of value  $b$ .

| 軸          | 設定した係数 $b$ の値 |
|------------|---------------|
| 15 度方向 x 軸 | 0.10481       |
| 15 度方向 y 軸 | 0.05638       |
| 45 度方向 x 軸 | 0.10481       |
| 45 度方向 y 軸 | 0.10481       |
| 75 度方向 x 軸 | 0.05638       |
| 75 度方向 y 軸 | 0.10481       |

値に設定した理由は、5.1.1 項で人がスマートフォンを移動させるときの平均的な移動距離は 25 cm 前後であることが見受けられたため、そのうちの x 軸成分もしくは y 軸成分の移動距離が 10 cm に近い値以下は非常に大きな誤差が含まれていることが図 3 より予想でき、そのような成分を排除したいからである。

### 5.2 カメラからマーカまでの距離推定

提案手法によるカメラからマーカまでの距離推定の精度を評価するため、提案手法と従来手法との比較実験について述べる。ここでの従来手法とは、xy 座標平面上の移動距離と、画面上に映った特徴点の移動距離を比較し、カメラから特徴点（マーカ）までの距離推定を行う手法である。

この実験では、AR マーカから 4[m] 離れた位置で、マーカを映すスマートフォンを被験者が手に持ちながら一振り動作を行う。使用した機器は表 1 にまとめている。被験者は静止した状態で各種センサのログ取得を開始し、一振り動作後に再び静止した状態でログ取得を終了する。その際、スマートフォンを図 5 に示すとおり xy 平面上での 15 度、45 度、75 度の 3 種の方向にそれぞれ 5 回ずつ移動させて（被験者 4 名が 5 回ずつ 3 方向で合計 60 試行）、提案手法と従来手法でカメラからマーカまでの距離推定をそれぞれ行い、その誤差割合を求めた。ローパスフィルタの係数は、予備実験の結果から、人がスマートフォンを移動させるときの平均的な移動距離は 25 cm 前後であることを考慮し、表 4 のとおりに設定した。なお、人が補助器具なしでスマートフォンを移動させることを想定しているため、この実験でも被験者 4 名が xy 平面上での 15 度、45 度、75 度の方向を意識してスマートフォンを移動させる方式でデータをとっているため、移動方向は正確ではなくばらつきがある。結果を図 4 に示し、そのときの各移動方向における誤差分布を図 6、図 7、図 8 に示す。

図 4 では、実際のスマートフォンからマーカまでの距離  $c_{real}$  に対する、提案手法や既存手法により測定されたスマートフォンからマーカまでの距離  $c_{estm}$  の誤差の占める割合  $100 \times \frac{|c_{real} - c_{estm}|}{c_{real}}$  を示している。これを測距の誤差割合と呼ぶ。移動方向が 45 度の場合は、提案手法と従来手法に違いはあまり見られない。これは、両手法ともに x 軸成分と y 軸成分の加重がおおよそ等しい値になっているため

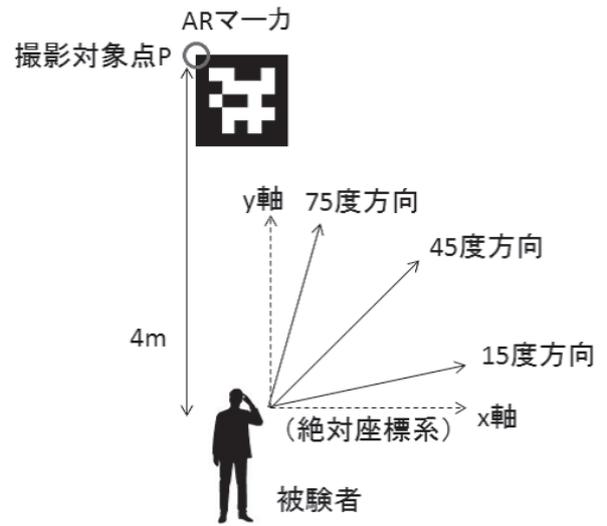


図 4 実験環境

Fig. 4 Experimental environment.

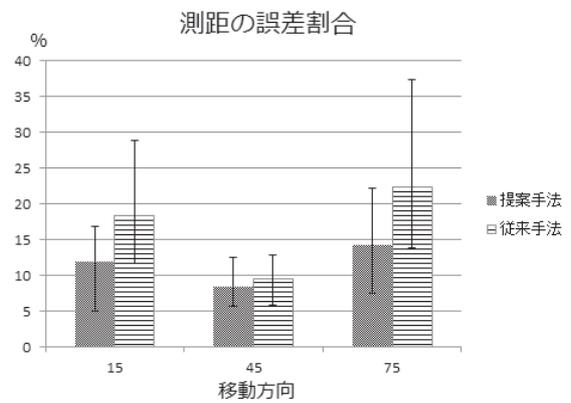


図 5 マーカまでの測距の誤差割合

Fig. 5 Error rate of shooting distance.

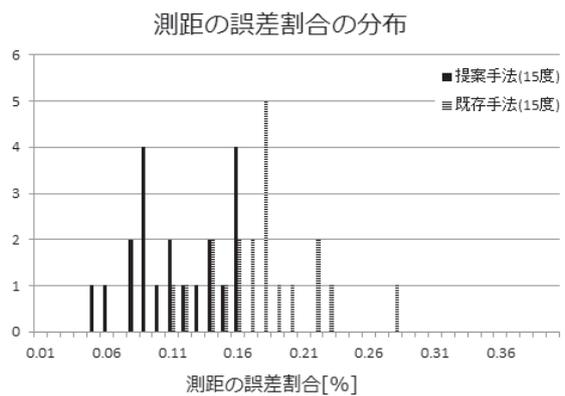


図 6 測距の誤差割合の分布 (15 度方向)

Fig. 6 Error distribution (direction: 15 degree).

である。移動方向が 15 度および 75 度の場合、提案手法の方が従来手法よりも平均的に約 8% 優れた性能を示している。移動方向 15 度での y 軸成分、移動方向 75 度での x 軸成分の移動量が少ないため、移動距離計測の誤差割合が非常に大きく、従来手法ではその影響を受けて誤差が増大する一方、提案手法では移動量が少ない成分の誤差を排除で

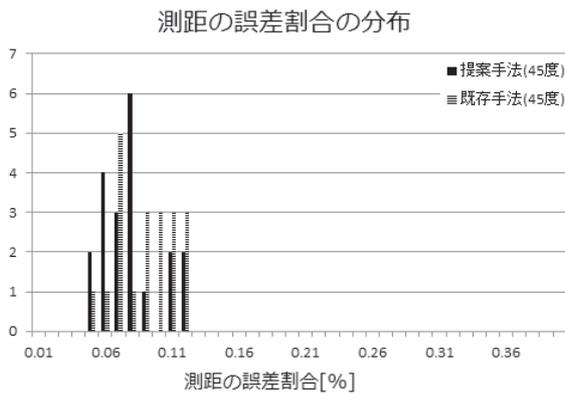


図 7 測距の誤差割合の分布 (45 度方向)

Fig. 7 Error distribution (direction: 45 degree).

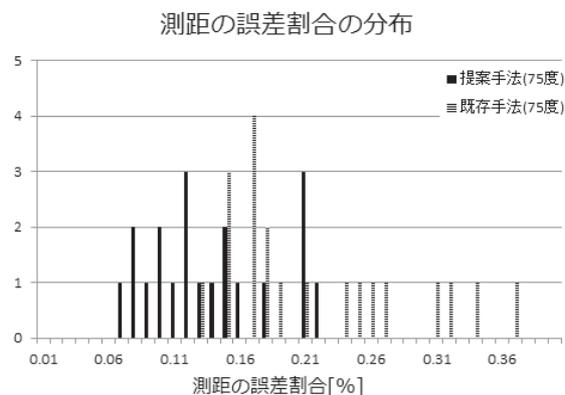


図 8 測距の誤差割合の分布 (75 度方向)

Fig. 8 Error distribution (direction: 75 degree).

きていることが理由としてあげられる。

さらに、図 4 で示す実験結果において、提案手法と既存手法の差異を調査するために t 検定を行った。提案手法と既存手法の 15 度方向、45 度方向、75 度方向について、各試行をサンプルとしたときの t 値はそれぞれ 5.91, 1.84, 5.13 であった。サンプル数 20 のときの有意 5% 水準の t 値は 2.09 であるため、提案手法と既存手法に関して 45 度方向では有意差は認められなかったが、15 度方向と 75 度方向においては有意差が認められる結果となり、提案手法の方が既存手法よりも高精度に測距できることが分かった。

図 6, 図 7, 図 8 では、移動方向が 15 度、45 度、75 度での提案手法と従来手法の各試行における測距の誤差割合の分布を示している。移動方向が 45 度の場合は、提案手法の方が誤差が小さくなってはいるが、ほぼ同等の結果である。その一方、移動方向が 15 度と 75 度の場合は、従来手法では測距の誤差割合が 25% を超えるなど、極端に悪くなっているケースがいくつか存在する。さらに、図 6, 図 7, 図 8 で示す結果において、F 検定により提案手法と既存手法における測距の誤差割合の分布を調査した。提案手法と既存手法の測距の誤差割合の分布について、F 値を求めた。その結果、15 度方向、45 度方向、75 度方向について、各試行をサンプルとしたときの F 値はそれぞれ 1.34,

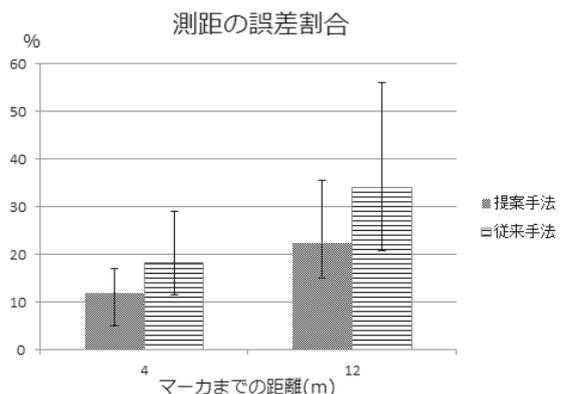


図 9 マーカまでの距離と測距の誤差割合

Fig. 9 Error rate of shooting distance from far away.

1.00, 2.03 であった。サンプル数 20 のときの有意 5% 水準の F 値は 2.12 であるため、提案手法と既存手法の測距の誤差割合の分布はいずれの移動方向においても等分散であるといえる。以上より、誤差の分布は同様でありながら、提案手法は既存手法よりも性能を向上させることができた。

今回の実験は 4 [m] の距離を計測しているため、測距の誤差割合が 25% を超えると 1 [m] より大きい誤差となってしまう。提案手法では、移動方向が 15 度の場合は最大約 16%、移動方向が 75 度の場合は最大約 22% に抑えることができています。

これらの結果から、3 種の移動方向をまとめて平均すると、提案手法は従来手法よりも約 6% 優れた性能を持つことを示すことができた。特に手に持ったスマートフォンを撮影対象物を映しながら移動させる場合、提案手法と従来手法ともに 45 度方向に移動させると精度が良くなることが分かった。さらに、15 度方向、75 度方向の場合においては、従来手法では精度が極端に悪化する場合があったが、提案手法では加重平均関数を導入して誤差の大きい成分を除去することによって、精度の低下を軽減させることができた。

### 5.3 カメラとマーカ間が遠い場合の測距の精度

本節では、スマートフォンとマーカ間の距離が遠い状態での測距の精度の調査を目的とした実験について述べる。カメラで映したマーカを認識できる限界距離が 12 m 程度であったため、スマートフォンとマーカ間の距離が 12 m の場合で、15 度方向にスマートフォンを移動させたときの提案手法と既存手法の測距の誤差割合を測定した。この実験に関しては、被験者は 1 名でスマートフォンとマーカ間の距離が 4 [m] のときと同じ回数だけ試行した。その結果を図 9 に示す。

図 9 より、提案手法は既存手法よりも高精度に測距できているが、測距の精度は 4 m のときに比べて全体的に悪化していることが分かる。その理由として、人がスマートフォンを所持しているため、静止状態であっても手ぶれ

によって加速度センサにノイズが入ったり、画面に映った AR マーカがぶれたりすることがあげられる。さらに、オートフォーカス機能付きのカメラを使用したため、環境によってはピントの調節の際に画面がぶれる現象が発生することも見受けられた。式 (14) および式 (18) において、スマートフォンとマーカ間の距離が遠いほど画面に映った AR マーカの位置変化量  $q_{x2} - q_{x1}$  および  $q_{y2} - q_{y1}$  が小さくなるため、画面に映った AR マーカのぶれの影響が大きくなる。そのため、精度の低下が起きてしまったと考えられる。以上より、スマートフォンとマーカ間の距離が遠い状態では、画面に映った AR マーカのぶれの影響を抑えなければ精度の悪化が起きてしまうため、それに対処する別の手法が必要であることが分かった。

## 6. まとめ

本研究では、手に持ったスマートフォンの一振り動作でのモーシヨンステレオ法において、スマートフォンの移動成分を 2 種類に分解し、移動距離計測の誤差割合の大きさに応じて移動成分の加重を変更することで、スマートフォンから AR マーカまでの距離推定の精度を上げる手法を提案した。予備実験においては、ローパスフィルタの係数を決定し、一振り動作ではスマートフォンの回転量は少ないことを確認し、加速度センサの計測値から移動距離を推定するには移動距離の増加にともなって測距の誤差割合が小さくなることを確認した。これらの予備実験の結果から加重平均関数を設定して、提案手法と xy 平面上での移動距離からモーシヨンステレオ法を用いる従来手法との比較実験を行った。その結果、4[m] の距離推定実験において、スマートフォンを横移動および縦移動した場合、提案手法の方が従来手法よりも平均的に約 8% 優れた性能を示し、極端な精度の悪化を軽減させることができたことを確認した。また、距離推定精度を良くする移動の仕方は、斜め 45 度方向へ移動させるということが分かった。

## 参考文献

- [1] 田中秀幸, 角 保志, 松本吉央: 正面からの観測でも高精度な姿勢推定が可能な AR マーカ, 研究報告グラフィクスと CAD (CG) 2012-CG-147(3), pp.1-5 (2012).
- [2] 本田智史, 高橋伊久夫, 吉田享子: ロービジョン向け店舗内ナビゲーションシステムの提案, 研究報告情報システムと社会環境 (IS) 2012-IS-119(6), pp.1-7 (2012).
- [3] Gerstweiler, G., Vonach, E. and Kaufmann, H.: HyMoTrack: A Mobile AR Navigation System for Complex Indoor Environments, *J. Sensors*, Vol.16, No.17, pp.1-19 (2015).
- [4] QPToolkit, available from (<http://kougaku-navi.net/QPToolkit/>).
- [5] Shibata, N. and Yamamoto, S.: GPGPU-Assisted Subpixel Tracking Method for Fiducial Markers, *J. Information Processing*, Vol.55, No.1, pp.19-28 (2014).
- [6] ARToolkit: Coordinate Systems, available from (<http://www.hitl.washington.edu/artoolkit/>

documentation/cs.htm).

- [7] Kato, H. and Billinghurst, M.: Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System, *Proc. 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR '99)*, pp.85-94 (1999).
- [8] Hochgatterer, M. and Holzmann, C.: Measuring Distance with Mobile Phones Using Single-Camera Stereo Vision, *Proc. International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pp.88-93 (2012).
- [9] Davison, A.J., Reid, I.D., Molton, N.D. and Stasse, O.: MonoSLAM: Real-Time Single Camera SLAM, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.29, Issue 6, pp.1052-1067 (2007).
- [10] Bayat, A.J., Pomplun, M. and Tran, D.A.: A Study on Human Activity Recognition Using Accelerometer Data from Smartphones, *Proc. 11th International Conference on Mobile Systems and Pervasive Computing (MobiSPC)*, pp.450-457 (2014).
- [11] Artal, R.M., Montiel, J.M.M. and Tardos, J.D.: ORB-SLAM: A Versatile and Accurate Monocular SLAM System, *IEEE Trans. Robotics*, Vol.31, No.5, pp.1147-1163 (2015).
- [12] Xia, H., Qiao, Y., Jian, J. and Chang, Y.: Using Smart Phone Sensors to Detect Transportation Modes, *J. Sensors*, Vol.14, No.11, pp.20843-20865 (2014).
- [13] Sensor Kinetics, available from ([http://www.rotoview.com/sensor\\_kinetics.htm](http://www.rotoview.com/sensor_kinetics.htm)).



勝間 亮 (正会員)

2006 年京都教育大学教育学部情報数学科卒業。2011 年奈良先端科学技術大学院大学情報科学研究科博士後期課程修了。同年より現在、大阪府立大学理学系研究科助教。ネットワークの研究に従事。