

# 全天球カメラを用いた机上物体に対する位置推定

増田 純也<sup>†</sup> 横山 昌平<sup>††</sup>

<sup>†</sup> 首都大学東京 システムデザイン学部 〒191-0065 東京都日野市旭が丘 6-6

<sup>††</sup> 首都大学東京大学院 システムデザイン研究科 〒191-0065 東京都日野市旭が丘 6-6

E-mail: <sup>†</sup>listxx775443@gmail.com, <sup>††</sup>shohei@tmu.ac.jp

あらまし 全天球カメラとは天球型に全方位の風景を撮影することのできるカメラであり、Google ストリートビューやVRなどに利用されている。近年、このカメラが安価なものになってきており一般のユーザーにも使いやすくなっている。研究においても全天球カメラを用いた物体検出が行われている。しかし、全天球カメラから物体の位置を推定する研究は少ない。また、全天球カメラで撮影した画像は正距円筒図法で変換されてパノラマ画像になるが、画像の上部下部で歪みが生じるため人間の目でその画像を見ても具体的な数値で画像の位置を推定することは難しい。そこで本研究では、全天球カメラから取得した物体の位置を上から俯瞰したXY座標で位置座標を取得することを行った。具体的には、物体検出手法であるYOLOv3を用いて検出された座標から現実の位置座標に変換する手法を用意した。この結果から、全天球カメラを入力として物体の位置情報を用いたアプリケーションの開発を行うことが可能だと考えている。

キーワード 全天球カメラ, 位置推定, 物体検出, YOLOv3

## 1 はじめに

近年、Virtual Reality(以下、VRと略す)やAugmented Reality(以下、ARと略す)の撮影にも用いられる全天球カメラと呼ばれる全方位の風景を撮影できるカメラが普及している。実用例としてはGoogle ストリートビューや不動産物件の内見の撮影など、さまざまな場面で用いられている。商品の種類も増えてきており、RICO社のTHETAシリーズ<sup>1</sup>をはじめとしたさまざまな商品が出てきている。その中でも安価な全天球カメラも販売され、一般のユーザーも使いやすくなっているため、Twitter<sup>2</sup>やInstagram<sup>3</sup>等のSNSで画像を共有する者が増えてきている。

全天球カメラで撮影して保存される画像の種類には全天球球面画像と全天球パノラマ画像の2種類がある。全天球球面画像はVRやARに用いられる球面画像のことである。専用のアプリケーションを用いることで実際にユーザーがその場にいるような感覚で画像を閲覧することができ、任意の方向を見ることができる。表示する画像は撮影した画像全体を見せるのではなく、ユーザーの視点に合わせて画像を切り出し、補正して見せている。欠点としては、球面画像になっているため1度にすべての風景を眺めることができず、特定の方向しか見ることができない。

それに対して全天球パノラマ画像は通常のカメラで撮影した画像と同じ形に全天球球面画像を変換した平面画像になっている。RICO社のTHETAシリーズでは正距円筒図法を用いて全天球球面画像を全天球パノラマ画像に変換している。全天球

パノラマ画像の特徴は、全方向の風景をまとめて見ることが可能になっていることである。また、全天球球面画像とは違い、閲覧するために専用のアプリケーションを必要としないため特別な準備をせずに画像を見ることができる。しかし、全天球パノラマ画像は球面画像を平面画像に変換しているため、全天球パノラマ画像の上部と下部で歪みが生じてしまい、人や物体などの画像内に写るものを正しく認識することが困難になってしまう。

その中でも我々は、全天球カメラで撮影した全天球パノラマ画像に着目した。最初に、全天球パノラマ画像の上部下部に対しても物体検出を行えるのかを確認するために物体検出手法であるYou Only Look Once version 3(以下、YOLOv3と略す)を用いて検証した。検出する物体は、直径3cm、高さ1cm、色がピンクとグレーの2種類で構成された円柱型の駒を用意し、YOLOv3で学習、検出を行った。検出結果を見ると、歪みがあるにも関わらず90%以上の確率で駒であるという検出結果が出た。このことから、全天球パノラマ画像の上部下部に移る物体でも構造が単純なモデルなら十分な検出結果が得られることがわかる。しかし、歪みが大きくなっている部分で物体検出をおこなえても、人の目で物体が具体的にどの位置にあるのかを把握することは難しい。

物体検出で用いた駒を写した全天球球面画像と全天球パノラマ画像を図1、図2に示す。図1は駒を見るために球面画像の下部分を切り出したものを表示しているため、駒がどの位置にあるのか大体把握することができる。それに対し図2は全方向の風景をまとめてみることができ、画像の下部に写る駒や中部に写る人を確認できる。しかし、この画像から画像の歪みで駒が具体的にどの位置にあるか、人との距離の関係などの位置関係を人の力で把握することは難しい。これらのことより、画像の上部下部の物体を正確に認識し具体的な位置を把握でき、

1: <https://theta360.com/ja/>

2: <https://twitter.com/>

3: <https://www.instagram.com/>

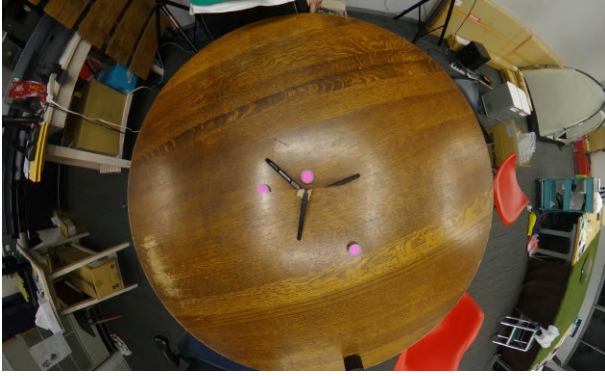


図 1 全天球球面画像の下方向を切り出した画像



図 2 物体検出で用いた駒を 3 つ写した全天球パノラマ画像

中部の歪んでない部分の物体検出と組み合わせることで全天球カメラの性能を十分に生かすことができると考える。

本研究では、全天球カメラを机の上に置き、机の上にある物体の位置推定手法を提案する。具体的には、全天球カメラで撮影した全天球パノラマ画像を、YOLOv3 を用いて事前に用意した学習データを使い物体検出を行う。検出された座標から推定される実空間での位置座標に提案手法を用いて変換する。推定された位置座標は、机を上から俯瞰した XY 平面上の値をとるようになっている。この手法を用いることで机上で行っている作業と、画像の中部に写る人との関係を組み合わせたアプリケーションの開発が可能になると考える。

本論文の構成は以下の通りである。2 章では、関連研究について述べる。3 章では、提案手法について述べる。4 章では、実験の結果と考察について述べる。5 章では、本研究のまとめと今後の展望について述べる。

## 2 関連研究

本章では関連研究について述べる。まず、全天球カメラを用いた研究をいくつか紹介する。中澤ら [1] は、全天球カメラをボールに内蔵し、それを用いて撮影した動画の視点を固定する手法を提案した。これによりスポーツ中継において競技場外からの撮影だけでなく競技場内からの撮影を可能にした。

佐々木ら [2] は、夜間の空の雲の様子を観察できるように全天球カメラを設定し、それらが VR 教材として教育現場で利用できるか検討した。前日の夜に撮影した雲を用いて生徒に次の日の天気を予測してもらい結果の天気をその場でわかるという点で 1 時間の授業での効率が上がると考えている。

このように全天球カメラを用いた研究は様々な分野で行われており、応用されている。その中でも複数の全天球カメラを用いた位置推定手法や運動推定手法を提案した研究があるので紹介する。

青木ら [3] は、カメラの自己位置推定と環境地図作図を同時に行う Simultaneous Localization and Mapping (SLAM) を全天球カメラで実現するための基礎検討を行った。2 つの全天球カメラを用いて、内部パラメータを使わずに外部パラメータを求め 3 次元再構成を行えるようにした。

野田ら [4] は、全天球ステレオカメラを用いた 3 次元計測結果を利用する運動推定の手法を提案した。ステレオカメラとの幾何的な関係から計測における誤差の大きい点を除去して運動推定の精度を向上させた。

また、本研究では小さい物体の位置推定を行う。小さい物体に似ているものとして指先の位置推定を行っている研究がいくつかあったので紹介する。

新谷ら [5] は、デバイスを持たずに、直感的に大型ディスプレイとのインタラクションを可能にするポスティングインターフェースを提案した。カメラから得た指差し画像のシルエットを抜き取り、それをもとに推定を行った。

Oka ら [6] は、赤外線カメラを上を設置して下にある机を撮影し、フレーム内の指先の位置を識別、指先の軌跡を測定する手法を提案した。人の体温に近い  $30^{\circ}\text{C}$  ~  $34^{\circ}\text{C}$  で手の領域を検出して指先と思われる部分をスコア付けして検出した。

これらの研究は、指先の位置推定を行うために特別なデバイスを用意している。

本研究では、特別な機材を設置せずに全天球カメラを机の上に置くだけで容易に位置推定を行えるシステムを構築した。また、複数の全天球カメラを用いずに、1 つの全天球カメラと高さのパラメーターも用いて実装した。

## 3 提案手法

図 3 に本研究の提案手法の流れを示す。はじめに、全天球カメラで駒を撮影し学習に必要なデータを集める。その後、集めたデータを用いて駒を学習する。学習データを用いて YOLOv3 で物体検出を行う。物体検出で得られた Bounding Box (以下、BBBox と略す) の座標から実空間の位置座標に変換する手法を用いて推定される座標を取得することを行った。

### 3.1 本研究の環境の定義

本研究では全天球カメラで撮影した全天球パノラマ画像に物体検出をし、その結果から実空間にある物体の位置を推定する手法を提案する。ここでは実空間の環境の定義、全天球パノラマ画像内の環境の定義、それらの環境に必要なパラメータの定義をそれぞれ説明していく。

#### 3.1.1 実空間の環境

まず実空間の環境について説明していく。今回想定する全天球カメラと駒の実空間での様子を図 4 に示す。このように全天球カメラを机の中央に、駒を任意の場所に配置してある。本研

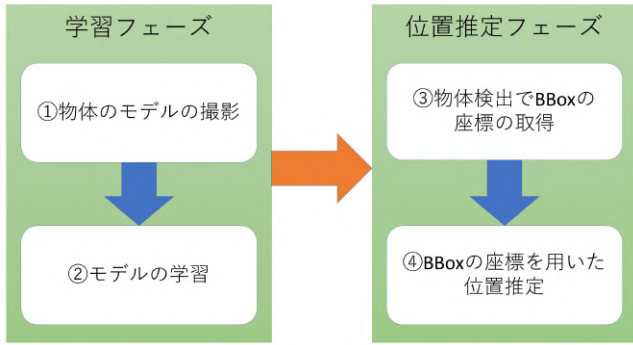


図 3 提案手法の流れ



図 4 実空間での全天球カメラと駒の状態

究ではこの任意に配置した駒に対して位置推定を行い机のどこに物体が置かれているのかを推定している。

本研究で用いた全天球カメラは RICOH 社の THETA Z1 [7](以下, THETA と略す) を用いた。THETA の外観を図 5 に示す。これを机の上に配置して位置推定を行っている。

駒と全天球カメラの位置関係を上から見た簡易図を図 6 に示す。図 6 に示すように、机の上を  $XY$  平面と定義する。その際、カメラの中心点から机に下した垂線と  $XY$  平面の交点を  $XY$  平面の原点とする。なお、THETA の右側を  $x > 0$ 、左側を  $x < 0$ 、裏側を  $y > 0$ 、表側を  $y < 0$  とする。また以下の内容で、実際の座標や推定された座標を  $(5, -5)$  で表現することがある。これは実空間で THETA の中心から THETA の右側に 5cm, THETA の表側に 5cm の位置に駒があることを意味している。

次に机上の極座標について説明する。ここでは原点から  $Y$  軸の負の値の方向に伸ばした半直線を机上の極座標の始線とする。原点から駒の中心点までの距離を  $r[\text{cm}]$  とする。始線から駒の中心点を時計回りに測った角度を  $\theta$  とする。

最後に横から見た全天球カメラと駒の位置関係について説明する。横から見た簡易的な位置関係を図 7 に示す。カメラレンズの中心点から原点までの高さを  $h[\text{cm}]$  とする。カメラレンズの中心点をとる机と並行な直線とレンズの中心点と駒の中心点を結ぶ直線のなす角を  $\Phi$  とする。

### 3.1.2 全天球パノラマ画像の環境

ここでは全天球パノラマ画像の環境の定義について説明して



図 5 THETA Z1 の外観 (画像の左から左側面, 表面, 右側面, 裏面)

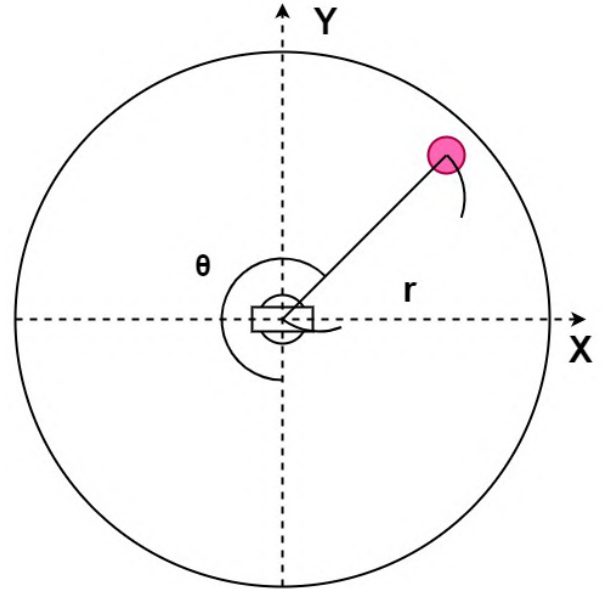


図 6 上から見たカメラと駒の位置関係

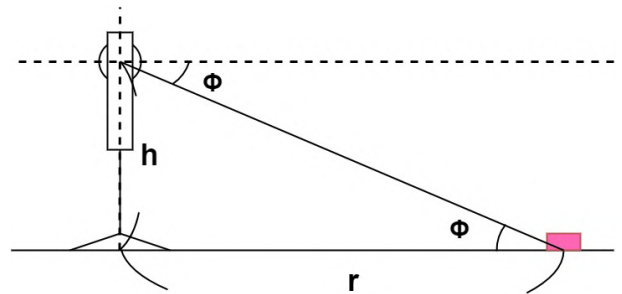


図 7 横から見たカメラと駒の位置関係

いく。なお、全天球パノラマ画像で現れるパラメータにはすべてダッシュを付けて表記する。物体検出後の全天球パノラマ画像の簡易的な図を図 8 に示す。図 8 に示してあるように全天球パノラマ画像での直交座標を左上を原点と置いた  $X'Y'$  平面として定義する。 $X'$  の値が増加すると実空間で説明した  $\theta$  が増加する方向の風景を見せている。 $Y'$  の値が増加すると  $\Phi$  の増加方向、つまり中心から下方の方向の風景を映し出している。また、全天球パノラマ画像の解像度を  $W' \times H'$  とする。

全天球パノラマ画像は実空間で定義した極座標の始線の位置から時計回りに風景を映し出している。 $\theta$  の定義を反時計回りの角度ではなく時計回りで定義したのはこのためである。

図 8 に物体検出後に駒の周りを BBox で囲まれている。BBox から与えられる情報には BBox の左上の座標  $(x', y')$ , BBox の



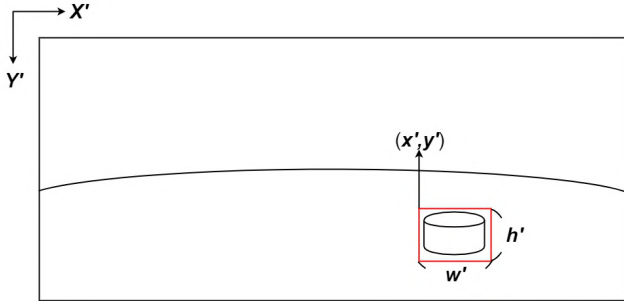


図 8 物体検出後の全天球パノラマ画像

表 1 THETA Z1 仕様

|                                 |  |
|---------------------------------|--|
| 幅                               | 48mm   |
| 高さ                              | 132.5mm  |
| 奥行き                             | 29.7mm   |
| 静止画像解像度                         | 6720 × 3360  |
| 動画解像度/フレームレート/<br>ビットレート        | 4K 3840 × 1920/29.97fps/56Mbps<br>2K 1920 × 960/29.97fps/16Mbps  |
| ライブストリーミング解像度/<br>フレームレート (USB) | 4K 3840 × 1920/29.97fps/120Mbps<br>2K 1920 × 960/29.97fps/42Mbps |

幅  $w'$ 、BBBox の高さ  $h'$  がある。位置推定を行う際にはこれらの情報を用いて進めていく。

### 3.2 物体のモデルの撮影・学習

本研究ではモデルの撮影に THETA を用いた。THETA の仕様を表 1 に示す。これを用いて静止画像の撮影を行う。また、物体のモデルは直径 3cm、高さ 1cm 色がピンクとグレー 2 種類で構成された円柱型の駒を 3D プリンターで作成した。

これらを机の上において全天球パノラマ画像の撮影を行った。撮影された駒はカメラとの距離  $r$ [cm] が小さくなるほど歪み方が大きくなっていく。本研究では画像に写っている机上の駒をすべて認識したい。そのため、カメラに対して様々な距離、方向からモデルを撮影する必要がある。

また、全天球パノラマ画像は実空間の始線の位置で駒が分断されている状態で写しだされるものがある。さらに、THETA では 2 つの魚眼レンズで撮影した画像をスティッチングと呼ばれる画像をつなぎ合わせる作業が行われている。そのため画像のつなぎ目の場所に駒が配置されていると、駒がぼやけて写る、駒の中心部が映らないなどの状態になる場合がある。図 9 に通常状態で写る駒と特殊状態で写る駒の画像を示す。これらに対しても認識する必要があるので撮影する。

このような特殊状態になる駒を含めて学習用の画像を集め学習をする。今回モデルの学習には 1 枚の全天球パノラマ画像に 3 つの駒が写っており、その画像を 218 枚用意し学習した。内訳としては始線の位置で分断された駒を 15%程、スティッチングでぼやけている、中央が映らない駒を 15%程、残りを通常状態の駒の割合で撮影した。また、始線で分断された駒は画像の右に写ったものと左に写ったものそれぞれ 1 つの駒として学習した。なお、学習には YOLOv3 を用いて 3000 回の学習を行った。



図 9 通常通り写る駒と特殊な状態で写る駒

### 3.3 物体検出・BBBox の座標を用いた位置推定

本研究では物体検出に YOLOv3 を用いた。今回学習させた物体は用意した駒のみで、出力結果には駒の情報のみが出力されるようになっている。物体検出の結果は物体のラベル名、確率、BBBox の左上の  $X'$  座標、 $Y'$  座標、BBBox の幅  $w'$ 、高さ  $h'$  が出力される。本研究では用意した駒のみを学習しているので、出力される座標、幅、高さの情報を抽出する。

BBBox の情報から机を上から俯瞰した  $XY$  平面上の値をとる座標  $(x, y)$  を推定する。図 6, 図 7, 図 8 で示したように BBBox の情報をそれぞれ  $x'$ ,  $y'$ ,  $w'$ ,  $h'$ 、カメラと駒の距離を  $r$ [cm]、カメラレンズの中心点と机との距離を  $h$ [cm]、実空間の極座標で始線から駒までを時計回りに測った角度を  $\theta$ 、カメラレンズの中心点を通る机と平行な直線とレンズの中心点と駒の中心点を結ぶ直線のなす角を  $\Phi$ 、全天球パノラマ画像の解像度を  $W' \times H'$  とする。これらの変数を用いて駒の中心の位置推定をする。なお、位置推定は BBBox の中心座標を用いる

BBBox の中心座標を  $(a', b')$  とすると、次式で求められる。

$$a' = x' + \frac{w'}{2} \quad (1)$$

$$b' = y' + \frac{h'}{2} \quad (2)$$

式 (1)、式 (2) で求めた  $a, b$  を用いて、実空間での  $\theta, \Phi$  を次式で表すことができる。

$$\theta = \frac{2a'\pi}{W'} \quad (3)$$

$$\Phi = \frac{(\frac{H'}{2} - b')\pi}{H'} \quad (4)$$

$\Phi$  と高さ  $h$ [cm] を用いて全天球カメラと駒の距離  $r$ [cm] は次式で表すことができる。

$$r = \left| \frac{h}{\tan \Phi} \right| \quad (5)$$

これにより、実空間での極座標の要素の  $r$ [cm]、 $\theta$  を求められた。極座標を直行座標に変換することで推定座標  $(x, y)$  が求まる。変換式は次式で表すことができる。

$$x = -r \sin \theta \quad (6)$$

$$y = -r \cos \theta \quad (7)$$

このことから高さ  $h$ [cm] の値がわかれば推定座標  $(x, y)$  を求めることができる。本研究では撮影の際に三脚と THETA のエ

クステンションアダプターを用いた。これらを合わせて机からカメラのレンズの中心までの高さを計測したところ 35.8cm であったので本研究ではこの値で位置推定を行っていく。

## 4 実験結果

本研究では推定座標の精度を確認するために、駒を (5, 0) や (5, 5) などの代表点に置き提案手法で求めた推定座標との誤差を求める実験、駒を指ではじいて直線の軌跡と推定座標の誤差を求める実験の 2 種類を行った。4.1 節では駒を代表点に置いた静止状態での推定座標の誤差を求める実験、4.2 節では駒を指で弾き動いた状態での推定座標の誤差を求める実験について述べる。

### 4.1 静止状態での推定誤差

今回の実験では全天球カメラに対して前後左右斜めの 8 方向に駒を配置し位置推定を行った。駒の撮影は 1 つの方向に対して 4 種類の距離で 10 枚撮影した。実際の駒の座標と推定された  $X$  座標と実際の座標の誤差の平均値、 $Y$  座標の誤差の平均値、実際の駒の位置と推定された位置の離れた距離の平均値をまとめた表を表 2 に示す。

表 2 を見ると (0, 5) や (20, -20) などの比較的カメラと駒の距離が近いときに推定座標との誤差が小さいことがわかる。またカメラと駒の距離が 40cm 以内の時は、(0, -5) の場合を除いて距離の平均誤差が 3cm 以内に収まっていることがわかる。この誤差は駒やカメラの配置を手作業で行ったことで生まれたと考えられる。それに対して (0, 80) や (60, 60) などの距離が遠いときの推定誤差が大きくなっていることがわかる。この誤差は手作業で配置したことによって生まれた誤差にしては大きすぎる。大きさとしては、40cm を超える位置だと 5cm 以上の誤差が生じている。このことから本研究の提案手法はカメラから 40cm 以内の近距離に対しては有効な手法であることがわかる。しかし、40cm を超える距離に対しては誤差が大きくなり正確な物体の位置を知るには適していない。

さらに、(0, -5) での誤差が他の方向の近距離の位置推定と比べて誤差が大きくなっている。これは、(0, -5) で撮影した駒が全天球パノラマ画像の左と右で分断されてしまったことが原因である。全天球球面画像から全天球パノラマ画像に変換する際にどうしても縦に分断する必要がある。その分断する場所が駒と重なってしまったため駒が 2 つに分断された状態で表示されている。さらに分断された駒の物体検出の結果が画像の左側に写る駒のみを認識していたため BBox の座標が本来あるべき座標の結果を得ることができなかった。

この対処法としては、駒が分断されている状態でのトレーニングデータを増やして左右に写る駒を認識できるようにする。そのうえで BBox の情報を受け取る前に 2 つの BBox を合併することで本来あるべき座標を取得することができると考えられる。もしくは全天球パノラマ画像の左右に同じ画像を結合させ、画像の幅を 2 倍にしても対処できると考えられる。ただし、この場合は画像の情報量が大きくなるので処理速度が遅くなる可

表 2 駒が静止状態での推定座標の誤差

| 実際の座標      | X 座標の誤差の平均値 [cm] | Y 座標の誤差の平均値 [cm] | 距離の誤差の平均値 [cm] |
|------------|------------------|------------------|----------------|
| (0, 5)     | 0.69652          | 0.44412          | 0.826139       |
| (0, 20)    | 0.887478         | 0.243602         | 0.920941       |
| (0, 40)    | 1.546079         | 1.631867         | 2.249701       |
| (0, 80)    | 2.206765         | 7.486312         | 7.808858       |
| (0, -5)    | 3.3695582        | 1.6911646        | 3.770295       |
| (0, -20)   | 1.558022         | 0.0597478        | 1.559479       |
| (0, -40)   | 2.3973815        | 0.329535         | 2.420155       |
| (0, -80)   | 2.600332         | 7.821876         | 8.394071       |
| (5, 0)     | 0.856195         | 0.334833         | 0.91993        |
| (20, 0)    | 1.53444          | 0.150271         | 1.54228        |
| (40, 0)    | 2.014947         | 0.696572         | 2.139083       |
| (80, 0)    | 5.364418         | 2.316764         | 5.846846       |
| (-5, 0)    | 0.209044         | 0.46940821       | 0.515042       |
| (-20, 0)   | 1.4482021        | 2.05897164       | 2.548836       |
| (-40, 0)   | 2.4040923        | 1.76629264       | 2.983805       |
| (-80, 0)   | 6.9977056        | 4.51099402       | 8.326649       |
| (5, 5)     | 1.184996         | 0.186634         | 1.199849       |
| (20, 20)   | 1.486037         | 0.399048         | 1.540412       |
| (40, 40)   | 3.345595         | 3.14901          | 4.596776       |
| (60, 60)   | 5.310618         | 4.577767         | 7.014227       |
| (5, -5)    | 0.230229         | 0.7650963        | 0.799167       |
| (20, -20)  | 0.490222         | 0.653413         | 0.820154       |
| (40, -40)  | 0.282809         | 1.9406759        | 1.968027       |
| (60, -60)  | 2.48815          | 4.24926551       | 4.926642       |
| (-5, 5)    | 0.585498         | 0.693247         | 0.907762       |
| (-20, 20)  | 0.699675         | 2.730901         | 2.819211       |
| (-40, 40)  | 3.352174         | 5.655845         | 6.575512       |
| (-60, 60)  | 6.963942         | 11.83793         | 13.73958       |
| (-5, -5)   | 1.1952767        | 1.467548         | 1.892879       |
| (-20, -20) | 2.0974588        | 0.387548         | 2.133519       |
| (-40, -40) | 2.6414438        | 0.630181         | 2.719949       |
| (-60, -60) | 5.074087         | 2.8805631        | 5.836568       |

能性がある。

また、表 2 でまとめた距離の誤差の平均値と誤差の方向を可視化した図を図 10 に示す。矢印の中心が実際の座標で、矢印の向きが誤差の方向、矢印の大きさが誤差の大きさを示している。なお、矢印の大きさは、距離の誤差が最も小さい (0, -5) を確認できるようにするためにすべての矢印の大きさを実際の値よりも大きくしている。

図 10 からカメラとの距離が大きくなるほど誤差が大きくなり、誤差の方向もカメラから離れる方向に向いていることがわかる。この原因としては BBox の中心座標を用いて位置推定を行ったためだと考えられる。つまり、レンズの中心と BBox の中心座標を結ぶ直線上にある机の位置を推定している。そのため、駒の高さが限りなく 0 に近ければ BBox の中心座標を用いて位置推定を行っても誤差が大きくならずに済むが、今回用いた駒は高さ 1cm あるので  $\Phi$  の値が小さくなるほど誤差が大きくなる。

この誤差を抑えるためには、事前に現れると思われる誤差の大きさと方向を予測して補正する方法が考えられる。静止状態の駒の位置推定の誤差を表 2 と図 10 で示した実験結果よりも細かく測定し、位置推定の後処理で補正することでより正確な推定結果を出すことが可能だと考えられる。

また、BBox の中心座標とは違う座標を用いて位置推定を行うことで誤差を抑えることが可能だと考えられる。 $\Phi$  の値が小さくなるほど誤差が大きくなっているため、 $\Phi$  の値が小さくなるほど  $y'$  を増加させた座標で位置推定を行うことで誤差を減らすことができると考えられる。

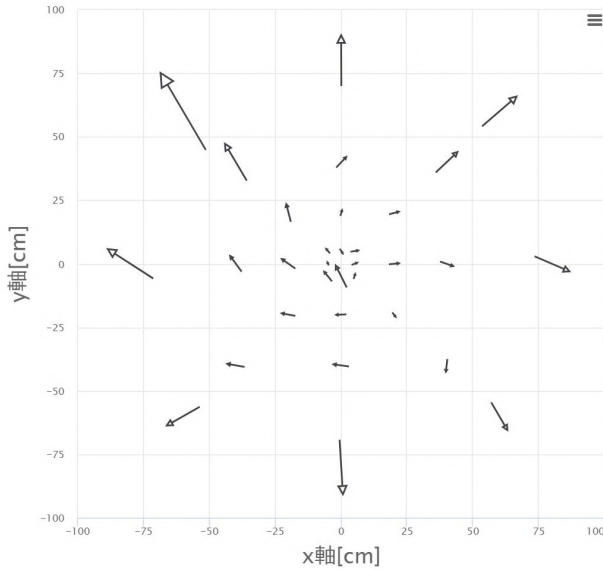


図 10 駒が静止状態の誤差の平均値

#### 4.2 動いている駒の推定誤差

ここでは駒を指で弾き動いている駒に対しての駒の推定座標と実際に通ったと思われる駒の軌跡との誤差を求めていく。実験の進め方は、駒を任意の場所に配置し任意の方向に指ではじく。その様子を全天球カメラで動画を撮影し全天球パノラマ動画を用いて位置推定を行う。この実験では 4.1 節で行った実験のように駒の正確な位置を知ることは難しい。そのため駒をはじく前とはじいた後の駒が止まっている状態をそれぞれ 5 秒ほど撮影しておき、その静止状態の駒に対しての推定座標の平均値を駒の始点、終点とし、始点と終点を結ぶ線分が駒が実際に通った軌跡であると想定する。また、停止状態の駒推定座標を確認すると 1 つ前のフレームとの距離の誤差が 0.2cm 以内であったのでこの実験では前フレームと比べて 0.2cm よりもずれていた場合、駒は動いている状態とする。

図 11 に駒をはじいたときの推定座標と駒が通った軌跡、表 3 に推定誤差の代表値を示す。表 3 から動いている駒の推定座標と実際に駒が通った軌跡との誤差は小さく、誤差の平均値が 4[cm] 以下という結果となった。そのため十分な精度で位置推定を行えたと考えられる。

また、図 11 を見ると始点の点群にむらがある。この原因は、駒をはじく際に手を近づけたことにより駒周辺の状態が変わってしまったことだと考える。動画の最初のフレームでは駒に手を近づけておらず、はじく寸前のフレームと比べると駒周辺の背景や輝度が変わっている。これにより BBox の座標の誤差が生まれている。動画の最初のフレームとはじく寸前のフレームの推定座標の距離の誤差を計算すると 0.5193cm であった。この対策としては物体のモデルを学習する際に様々な輝度のデータや駒の周辺の背景を変えたデータも用いて学習することで誤差を押さえることができると考えられる。

また、THETA の裏側 ( $y > 0$ )、左側 ( $x < 0$ )、表側 ( $y < 0$ )、右側、( $x > 0$ ) の 4 方向の範囲で駒をはじいた動画をそれぞれ 20 本撮影した。駒の動きとしては大きくうごいたもの、小さく

表 3 誤差の代表値

| 平均値 [cm]    | 最大値 [cm]    | 最小値 [cm]    |
|-------------|-------------|-------------|
| 0.378633067 | 0.589704676 | 0.144443004 |

動いたもの、THETA に向かっていくもの、THETA から離れていくもの等様々な動きの様子を撮影した。それらの動画に対して誤差を測定し、20 本の動画で生まれた誤差の平均値を求めた。各動画に対する誤差の平均値を表 4、表 4 で示した誤差の各方向の平均値を表 5 に示す。

表 5 を見ると、各方向の誤差が 0.5cm 以下に収まっていることがわかる。そのため、どの方向に駒を動かしても位置推定には影響はないと考えられる。また、表側と裏側の誤差に対して左側、右側の誤差が大きくなっていることもわかる。これはスティッチング処理によって駒がぼやける、一部が写らない状態が動画中に含まれていることが原因だと考えられる。この対処法としては、物体のモデルの学習の段階でスティッチング処理によって通常の状態で写らない駒のデータ数を増やすことで誤差を減らすことが可能だと考えられる。

また今回の実験結果では表側が一番誤差が小さくなった。これは表 2 と図 10 で示した誤差の大きさが関係していると考えられる。表 2 と図 10 を見ると、THETA の表側に置かれた駒に対する位置推定の誤差がほかの範囲と比べて小さい傾向がある。そのため、表側ではじいた駒に対しても誤差が小さくなったと思われる。

また、表 4 を見ると誤差の平均が 1cm を超えている動画は 2 本のみになっている。今回の実験で用いた動画は全部で 80 本なので、97 % 以上が誤差の平均値が 1cm 以下という結果となった。そのため駒をはじく大きさ、向きに関係なく動いている駒に対して十分な精度で位置推定が行える。

## 5 まとめ・今後の課題

本研究では全天球カメラを用いて机の上にある物体の位置推定を行った。物体検出には YOLOv3 を用いて行い、THETA を用いて駒の撮影をし全天球パノラマ画像で学習をした。位置推定をする際は実空間、全天球パノラマ画像の各パラメータを用いて推定された座標を計算する。今回提案した手法では高さ hcm を事前に与えることで位置推定を行うことができる。

静止している物体に対する位置推定の実験からは、カメラとの距離  $r$ [cm] が大きくなるほど誤差が大きくなることが分かった。カメラとの距離が 40cm 以内であれば誤差が 3cm 以下で位置推定を行うことができた。

動いている物体に対する位置推定の実験から、カメラに対して 4 つの方向で誤差の平均を求めた。結果はどの方向も 0.5cm 以下という結果が得られた。また、撮影した動画のうち 97% 以上が誤差の平均値が 1cm 以下という結果となった。このことから、動いている物体に対しても提案した手法は有効だと考えられる。

今回の課題としては、物体が遠くに離れた際に推定座標の誤差が大きくなってしまうことである。原因としては位置推定の

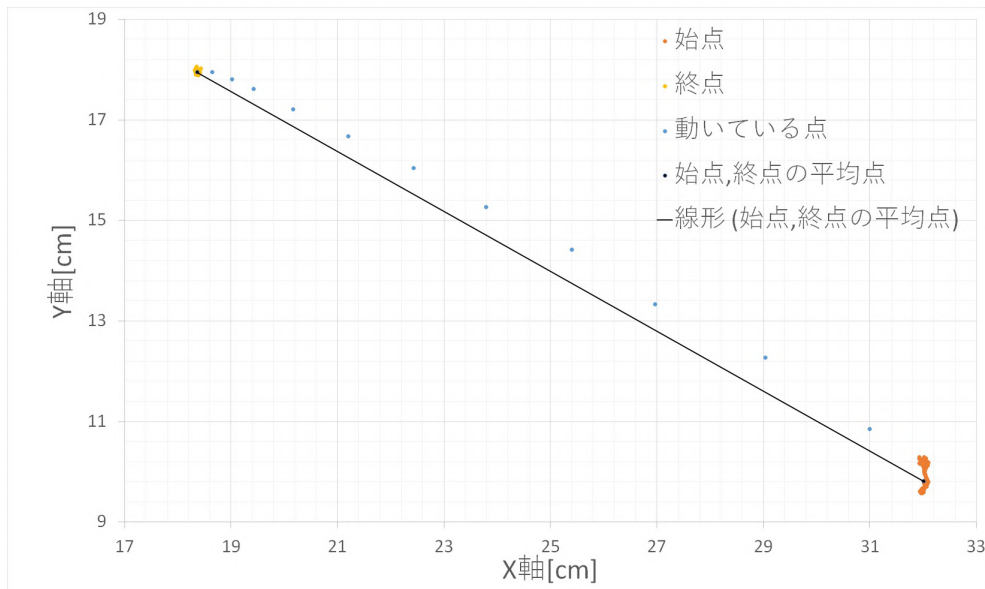


図 11 駒をはじいたときの駒の位置推定座標

表 4 各動画に対する誤差の平均値

| 裏側 [cm]  | 左側 [cm]  | 表側 [cm]  | 右側 [cm]  |
|----------|----------|----------|----------|
| 0.346861 | 0.550531 | 0.087333 | 0.605678 |
| 0.604328 | 0.692395 | 0.211361 | 0.134349 |
| 0.159286 | 0.3424   | 0.123878 | 0.735142 |
| 1.079    | 0.268691 | 0.163833 | 0.226793 |
| 0.63534  | 0.519812 | 0.158157 | 0.288261 |
| 0.131455 | 0.757626 | 0.314224 | 0.566506 |
| 0.558364 | 0.93346  | 0.547048 | 0.125717 |
| 0.343491 | 0.420624 | 0.069249 | 0.242004 |
| 0.814331 | 0.517864 | 0.083693 | 0.107736 |
| 0.066046 | 0.396695 | 0.08272  | 0.894708 |
| 0.061045 | 0.465695 | 0.501361 | 0.317555 |
| 0.114182 | 0.119757 | 0.193013 | 0.34415  |
| 0.184871 | 0.319564 | 0.1076   | 0.230432 |
| 0.171448 | 0.131276 | 0.372295 | 0.523302 |
| 0.109217 | 0.206088 | 0.207379 | 0.473652 |
| 0.146127 | 0.43495  | 0.212443 | 0.77889  |
| 0.264747 | 0.36992  | 0.16198  | 0.604959 |
| 0.405    | 0.260933 | 0.130713 | 0.16761  |
| 0.256411 | 0.261625 | 0.256819 | 0.625862 |
| 0.316302 | 0.391483 | 0.393298 | 1.058165 |

表 5 表 4 で求めた誤差の各方向ごとの平均値

| 裏側 [cm]  | 左側 [cm]  | 表側 [cm] | 右側 [cm]  |
|----------|----------|---------|----------|
| 0.338393 | 0.418069 | 0.21892 | 0.452573 |

計算の際に BBox の中心座標を用いているためであると考えられる。推定座標を見ると、カメラと駒の距離が離れると駒と同じ方向の誤差が大きくなっていた。つまり、カメラレンズの中心点と BBox の中心を結ぶ直線上にある机の座標を推定座標としたため  $\Phi$  の値が小さくなるにつれて誤差が大きくなってい

ると考えられる。そのため BBox の座標によって位置推定に用いる座標を変える必要がある。

また、本研究では高さを既知のものとして扱ったが、実際には事前に知っている場合は少ない。そのことから高さ推定を行う必要もあると考える。Iro Laina ら [8] は、単眼のカメラで撮影した画像の深度マップの推定を行った。この手法を全天球カメラに対応することができれば、事前に知る必要のあるパラメータのない位置推定を行うことができると考えられる。

## 文 献

- [1] 中澤正和, 小池英樹. “全天球カメラ内蔵ボールにおける視点固定手法”, 日本バーチャルリアリティ学会論文誌, 2017, 22. 4: 485-491.
- [2] 佐々木恒, 名越利幸 “VR を用いた夜間の雲の観察に関する教材開発”, 日本科学教育学会研究会研究報告, 2018, 33. 1: 129-132
- [3] 青木貴大, 渡辺裕, 原潤一 “全天球画像を用いたカメラ自己位置推定技術と環境地図作成の同時処理技術 (Simultaneous Localization and Mapping Using Omnidirectional Camera)”, 2018 年映像情報メディア学会冬季大会, 11D-4, Dec. 2018
- [4] 野田 純平, Sarthak Pathak, 藤井 浩光, 山下 淳, 浅間 一: “計測点の信頼度を考慮した全天球ステレオカメラの運動推定”, 精密工学会誌, Vol. 85, No. 6, pp. 568-576, June 2019.
- [5] 新谷晃市, 間下以大, 清川清, 竹村治雄 “大画面ポスティングシステムのための回帰モデルによる単眼画像からの指差し位置の推定” 情報処理学会研究報告, Vol. 2009-CVIM-167, No. 33, 2009.
- [6] Kenji Oka, Yoichi Sato, Hideki Koike “Real-Time Fingertip Tracking and Gesture Recognition” IEEE Computer Graphics and Applications, Volume:22 Issue:6, 2002
- [7] 製品紹介 — RICOH THETA Z1. <https://theta360.com/ja/about/theta/z1.html> accessed: 2019-12-27
- [8] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, Nassir Navab, “Deeper depth prediction with fully convolutional residual networks.” 2016 Fourth international conference on 3D vision (3DV). IEEE, 2016.