

画像認識の開発と応用

3年A組 中盛 楓也

3年A組 小林 望

3年B組 望月 草馬

指導教員 藤野 智美

1. 要約

私たちはハードウェアに搭載することを目的とした画像認識システムについて研究を行っている。研究の第一段階として、形や色が単純であるピンポン球の認識を試みた。その手法は、webカメラでピンポン球を映し、その映像データを Processing と OpenCV を用いて解析することで色や形、座標を求めるというものである。さらに、これらの技術を使用し、画像認識システムを用いた「移動可能型 VR」の開発を行ったので、ここで報告する。

キーワード 画像認識、webカメラ、物体の色と形、VR

2. 研究の背景と目的

私たちは画像認識システムに興味をもち、研究を開始した。今回はロボコン等で多用されているピンポン球を認識し、座標を求めるシステムの開発を行った。また、ここで得た画像認識の技術を使って VR への応用が可能なのではないかと考え、移動可能型 VR への応用を試みた。

3. 研究内容

3. 1 画像認識システムの必要な要素

今回目指した画像認識システムは、webカメラから得られた画像データのみから物体を認識し、座標を特定できるシステムである。この技術を用いると、画像認識システムを VR に応用する際、コントローラーによる操作無しで人間の動き（人間の位置情報）が感知できると考える。そこで、webカメラに物体を写し、得られた画像データのうち、「物体の色」、「物体の形」、「物体の座標」の3要素に着目して目的とする物体

のみを認識できる画像認識システムの開発を試みた。認識を行う物体には、色合い及び形状のシンプルさを考え、オレンジ色のピンポン球を用いることとした。

3. 1. 1 色の認識

オレンジ色のピンポン球は、単色であり、かつ環境色として少ないという点でその色に特徴がある。この色を正確に認識できるシステムが開発できれば、色の認識からピンポン球を特定できる可能性がある。プログラム上で以下の方法を組み込み、webカメラに写し出されたピンポン球の色の認識を試みた。なお、プログラミング言語には processing というオープンソースプロジェクトと、processing の追加ライブラリである OpenCV を用いた。

<方法>

1. webカメラでピンポン球を写し、画像データを得る。
2. 各ピクセルがオレンジ色かどうかを確かめるために、1で得られた画像データの

色の判定に用いる条件式を以下のように決める。

○オレンジ色の判定を行う条件式

<R,G,B>の色の値に対して、

$$R > G \times 1.5 \text{ かつ } R > B \times 1.5$$

なお、この条件式は、webカメラに様々なオレンジ色の物体を写し、それらの画像から得られたピクセルデータを分析して割り出した条件式である。つまり、この条件を満たす物体は、オレンジ色であると判断できる。

3. 1で得られた画像データの画面上のすべてのピクセルを2の手法を用いて判定し、オレンジと判定されたピクセルの座標を平均化する。この平均座標をピンポン球の中心として認識する。

<結果>

図1はプログラムの実行画面である。なお、オレンジ色と認識しているピクセルは黒で表示されており、ピンポン球の中心位置を表しているのが白い正方形である。この手法を用いた結果、オレンジ色である物体すべてに反応してしまい、ピンポン球だけを認識することはできなかった。また、図1のようにオレンジ色の物体が複数ある場合、座標の平均を算出しているため、それらの物体の間をさしてしまう。

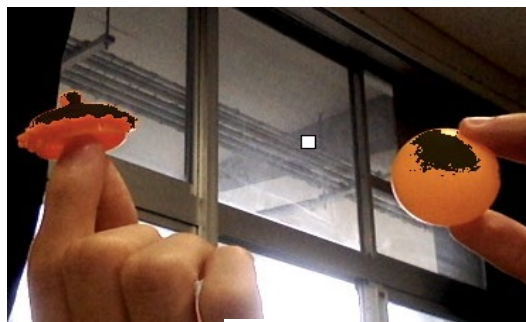


図1

<考察>

オレンジ色であるという条件だけではピンポン球のみを認識することができなかった。よって、色以外の他の要素を追加することでピンポン球の認識を行う必要がある。

3. 1. 2 形の認識

先述したように、色の認識のみではピンポン球だけを認識することはできなかった。そこで、色の認識に加えて、形を認識することができれば、ピンポン球以外のオレンジ色の物体を除外することができるのではないかと考えた。そこで、研究の次段階として、ピンポン球の円という形に注目し、プログラム上で以下の方法を組み込み、ピンポン球の形の認識を試みた。

<方法>

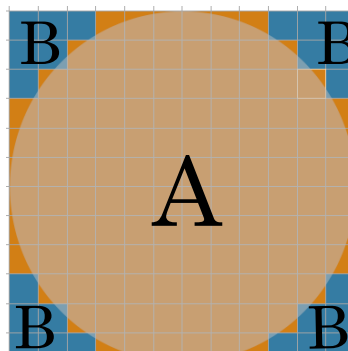


図3

1. 先述した色の認識の手法を利用し、画像のピクセルを上から順番に横一列にオレンジ色かどうかを確かめる、そして、オレンジが何ピクセル連続しているか記

録する。

2. 1で求めたオレンジ色の連続したピクセルの数が一番多いところを円の直径と定義し、この直線から円の中心と半径を求める。

3. 2で求めた円の中心から上下左右に半径の長さを伸ばし、正方形を作る。正方形の中に円があると仮定すると図2のように、正方形内で場所によってオレンジのピクセルの数が変わる。四隅付近はオレンジ色ではないが、中心付近はオレンジ色になっている。そのため、ピンポン球の判定条件として、正方形の四隅の角付近にはオレンジが少なく、中心付近にはオレンジが多いという条件を設定する。図2の領域Aにおけるオレンジのピクセル数が5割以上、領域Bにおけるオレンジのピクセル数が5割未満であれば、これを円と認識し、ピンポン球として判定した。

<結果>

この手法を用いると、オレンジ色の物体全てに反応することなく、ピンポン球のみに反応するようになった。また、図3のように複数のオレンジ色の物体がカメラに写っても、ピンポン球のみを認識することができた。しかし、オレンジ色で円形の物体であれば、ピンポン球でなくても反応してしまうことがわかった。

<考察>

確実にピンポン球のみを認識することはできないが、webカメラ一台のみでは平面でしか物体を認識することができないためこれ以上の認識精度向上が望めないと考えた。なお、ここまでの色の認識と形の認識によって、平面での物体の特定と認識は概ね可能になった。

3. 1. 3 座標の認識

次に、認識したピンポン球の座標を求める方法を考える。図4はカメラとピンポン球の位置関係を上から見たものである。

この考え方では、webカメラとピンポン球の距離をカメラの正面方向の距離(a)とwebカメラから見たピンポン球の左右方向のずれ(c)によってピンポン球の座標のずれを算出しようと考えている。

上図の a と r の関係式の算出方法として、webカメラとピンポン球との距離の係数に注目した。ピンポン球をwebカメラに近づけたり遠ざけたりすると、画面上に写し出されたピンポン球の大きさと実際のピンポン球の大きさの比率が変化する。このような遠近法の考え方から、 a を求められるのではないかと考えた。以下にその方法を示す。なお、プログラム上で、先述している色および形の認識のシステムを適用した上で、以下の方法を追加している。

①webカメラとピンポン球との正面方向の距離の算出 (a と r の関係式の算出)

<方法>

1. webカメラでピンポン球を写し、画像データを得る。
2. webカメラとピンポン球の正面方向の距離(a)と画面上のピンポン球のサイズ(r)

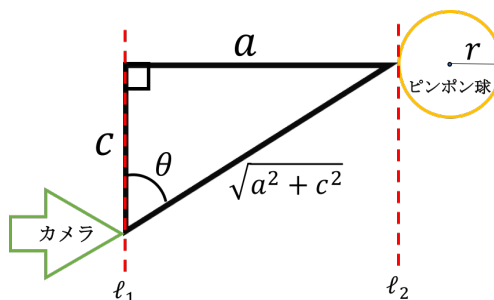


図 4

との関係を見つけるために、webカメラと

ピンポン球の間の距離をメジャーを用いて実測する。

3. この値を様々に変化させたとき、画面上のピンポン球の大きさがどのように変化するかを記録する。この段階で、いくつかのデータから、これらが反比例の関係になっていることが予測できた。よって、以下の手順では反比例の比例定数を導くことにする。

4. 3で予測した反比例の関係が成り立っていることを確かめるために、webカメラとピンポン球の正面方向の距離が15.00cmと30.00cmの2パターンで画面上のピンポン球の半径を測定し、比例定数の値を求める。

5. 4で求めた比例定数を反映させ、 a と r の関係式を算出する。この関係式の妥当性を検証するために、 a の値を15.00cmから2.00cmずつ増やし、 r の値を測定することで、理論値と実験値の差を検証した。

<結果>

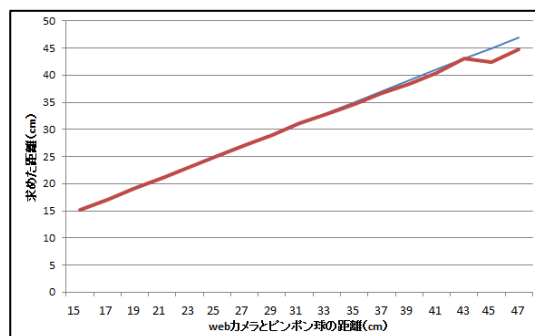
- webカメラとピンポン球の距離が15.00cmのとき、画面上のピンポン球の半径は82ピクセルであった。
- webカメラとピンポン球の距離が30.00cmのとき、画面上のピンポン球の半径は41ピクセルであった。

これらの結果より、webカメラとピンポン球との距離を a (cm)、画面上のピンポン球の半径を r (px)としたとき(図4参照)、以下の関係式が成り立つことが分かった。

$$a = \frac{1230}{r}$$

この関係式をプログラムに代入して a の値を変えながら r を測定した結果が図5である。なお、グラフ上の青線が理論値、赤線が実測値である。

なお、距離 (cm) の有効数字は小数第2位までとする。



距離 (cm)	ピクセル直径	$\frac{2460}{2r}$
15	163	15.09202
17	144	17.08333
19	128	19.21875
21	117	21.02564
23	107	22.99065
25	98	25.10204
27	91	27.03297
29	85	28.94118
31	79	31.13924
33	75	32.8
35	71	34.64789
37	67	36.71642
39	64	38.4375
41	61	40.32787
43	57	43.15789
45	58	42.41379
47	55	44.72727

図 5

<考察>

結果からわかるように、webカメラとピンポン球の距離 a と、画面上のピンポン球のサイズ r の関係を示す反比例の式を定義することができた。比例定数の値は、実験結果から1230であると定義でき、これはカメラによって変わる値であると考えられる。この関係式を用いれば、今回の実験とは逆の流れとなる、画面上のピンポン球のサイズの情報から逆算して、webカメラからどれだけ離れた位置にピンポン球があるかを算出できることになる。

②webカメラとピンポン球との左右方向のずれの算出 (a と c の関係式の算出)

次に、webカメラとピンポン球の正面方向の距離を変化させたとき、左右方向のず

れがどのように変化するか（図4のc）を以下の方法から求める。

<方法>

1. webカメラとピンポン球の距離をメジャーで測定し、20.00cmになるようにピンポン球を置く。
2. 1の状態からピンポン球の位置を左右に6.00cmずらし、画面上のピンポン球の座標がどれだけずれたかを計測する。
3. webカメラとピンポン球の距離を変化させ30.00cmになるように設定し、2と同様の作業を行い、ピンポン球の座標のずれを計測する。
4. 2, 3の結果から関係式を予想し、webカメラとピンポン球の距離(a)を変えながら、画面上での横ずれの値(c)を計測して、求めた関係式と一致するかを確認した。

<結果>

実験を行ったところ、ピンポン球の形が楕円になっていることに気づいた。これは、webカメラが広角レンズのため、画像のふちに近づくほど、物体が横に広がって写ってしまうことが原因である。つまり、ピンポン球がwebカメラの正面から左右にずれていればずれているほど、画面上での横の半径が大きく表示され、楕円となってしまうことがわかった。

<考察>

導いた関係式を適用するには、距離測定の方法を変更する必要がある。先ほどは、ピンポン球の横幅から半径を求めていたが、縦の幅から測定しなければならないと考える。

3. 2 VR への応用

私たちは開発した画像認識システムを

VRに应用できるのではないかと考えた。しかし、開発したシステムを用いて複雑な形状である人間の形を認識するのは困難である。そこで、先述した形の認識以外の色の認識や座標の認識の技術をVRに連動させて、「移動可能型VR」の開発を試みた。

開発した画像認識システムを活用することで、webカメラのデータのみからプレイヤー（体験者）の現在座標を求め、VR内のキャラクターに仮想空間内の座標を適用させることを試みた。それらを実現するための画像処理の手法として、背景差分法と特殊検出法を活用した。開発はゲーム開発エンジンの一つであるUnityで行った。なお、ライブラリ等は用いていない。

3. 2. 1 背景差分法

背景差分とは観測画像と事前に取得しておいた画像を比較することで、事前に取得した画像には存在しない物体を抽出する画像処理の方法である。この考え方をういてプログラムを作成した。

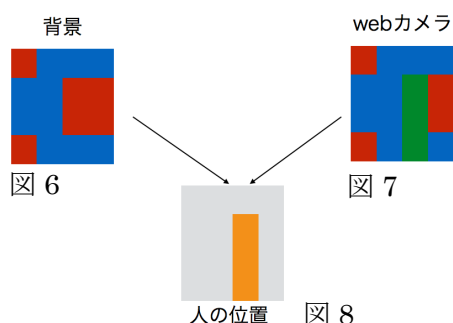
<方法>

1. 背景差分のプログラムを起動後、最初に背景を登録する。webカメラの画像に人を含む物体が写らないようにし、全ピクセルの色を記録し、これを基準の背景とする。なお、色の認識には、3.1.1で開発した画像認識システムを活用した。（図6）
2. その後、毎フレームで、登録した背景と現在の画像の、色が変わっているピクセルを見つける。この条件は、 $\langle R,G,B \rangle$ のいずれかの値において、背景と現在の映像と差が50以上ある値があれば、色が変わっていると判定する。（図7）
3. 画像内に写るものを人間に限定すれば、

背景と異なる色の集合は全て人間と判定できる。

4.2で「色が変わった」と判定されたピクセルの x 座標の平均を、人間の x 座標とする (図 8)。この座標の算出方法は、3.1.3の座標認識での手法と類似している。

この方法で人間の x 座標を求めることができる。今回は x 座標のみの算出を行ったが、カメラを2台用意すれば、人間の x 座標と z 座標を求めることができる。



<結果>

Unity で、web カメラの画像を参照し、各ピクセルの色を求め、画面上の全てのピクセル (フルピクセル) で処理すると、約 1 秒の処理時間を有してしまっただけで、実際には使用し、処理時間の少なかった画素数を 64 分の 1 のみ調べるといった方法をとった。しかし、結果として検出精度が低下してしまい、仮想空間内と現実空間での多くの誤差が発生し、うまく作動しなかった。よって Unity で背景差分の方法をとり、VR 映像を構成して体験させることはほとんど不可能であると判断した。

3. 2. 2 特徴点検出法

3.5.1 の背景差分法の問題点を解決するために、体に色のパターン (模様) の付いたマーカを取り付け、そのマーカを web カメラで見つけるという方法をと

った。この方法を特徴点検出法という。今回使用したマーカは上から赤、緑、赤の順に色分けされたもので体に巻きつけて使用する (図 10)。

<方法>

1. 画素数 $1/64$ で図 9 の矢印のように、左上から下に向かって順番にピクセルを調べ、色の認識を行う。
2. 赤、緑、赤の順に連続している列を検出する。なお、このパターン検出は、3.1.2 で検出した形の認識の手法を応用している。
3. 2 で検出した列の x 座標の平均を取ることによってマーカを中心の x 座標を認識する。

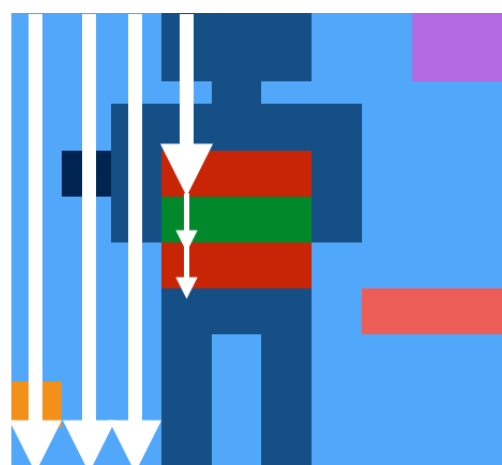


図 9



図 10

<結果>

この手法の処理時間は0.1秒以下となり、背景にマーカールと同じ色パターンがなければ必ず認識され、スムーズに処理されていた。ここで、求めた座標を、Unityのオンラインゲーム用サーバー「UNET」(図11)を経由してスマートフォンにデータを送ると、0.9秒のラグが生まれた。このラグの問題を解決するために、ほかの通信方法を使用する必要がある。

3. 3 UDP 通信

3. 3. 1 UDP 通信の導入

マーカールを装着する際の手間を無くすため、背景差分法をもう一度検討したが、Unityでの画像認識は負担が大きく、1280×720のピクセルの色を調べるという重い処理はUnityでは不可能であった。そこで、この問題を解決するために、処理の分担を行うことにした。

具体的には、画像処理に向いているprocessingで画像認識を行い、映像に写っている人間の座標を特定し、そのデータをUnityに送信してVR映像を構成するというものである。この方法をとるためには、processingとUnityの間で通信を行う必要がある。そこで、UDP(User Datagram Protocol)通信という方法をとることにした。UDP通信とは、リアルタイム性を求めた、IPプロトコルである。この方法を使うと、違うアプリケーション間で変数の共有を行うことができる。

3. 3. 2 UDP 通信の実装

次にコンピューターでこの背景差分プログラムを使って求めた座標をUDP通信を使ってスマートフォン上のUnityのアプリ

内に送信するシステムを開発した。(図11)

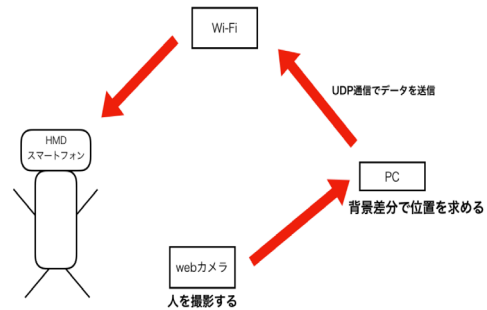


図 11

<方法>

1. スマートフォンとPCを同一のWi-Fiにつなぐ。
2. Processingを用いて背景差分法を行って求めた人の座標データを、UDP通信を使ってUnityに送信するシステムを作る。
3. Processing追加ライブラリ「udp」を使ってスマートフォンのローカルIPアドレスを指定してデータを送信する。
4. Unity上では、udp通信ライブラリを使用し、送られてきたデータをバーチャルワールドに適用してVR映像を流すようにする。

<結果>

この方法を用いると、作業をprocessingとUnityで分割することができた。結果として画像認識のパフォーマンスを確保したままVR映像を作成することができ、マーカール無しで人間の座標を測定できた。しかし、UDP通信を使うことでタイムラグを減少できたが、平均約0.2秒のタイムラグが発生した。

<考察>

プログラムを実行してわかった問題点は、以下である。

- 1.カメラを2台用意しないと人のx,z座標を求めることができない。
- 2.仮想空間内で移動可能な広さと同じ大きさの部屋を用意しなければならない。

問題点1の解決策として、足の裏の位置のy座標を求めることで、カメラと人の奥行きを距離を求めることができると考えられる。ラグの問題点として、UDP通信はデータの損失を確認しないことで速さを実現するというIPプロトコルだが、それでもVRで体験するには十分違和感が発生するタイムラグとなっている。よって、ネット回線または別の通信方法、ソフトの軽量化を考えなければならない。

4. 考察

Webカメラの画像のみから、<R,G,B>の値と、それらピクセルの配置からピンポン球を認識することに成功し、ピンポン球の奥行きを座標を求められた。しかし、広角レンズでの横ズレを求めることができていない。

また、VRにおいて、UDP通信を使うことで、処理の分担と背景差分の開発に成功し、歩行可能VRを実現できたが、ラグがまだまだ問題となっている。

5. 今後の展望

ピンポン球の認識では、光の強さや当たり方によって反応精度が低下してしまうという問題点がある。この問題の解決のために、光の当たり方を踏まえたアルゴリズムを組む必要がある。今回使用したピンポン球は単色で、球のためどの方から見ても円

に見えると、単純な形をしている。そのため、もっと複雑な形状、色彩を持つ物体を識別するアルゴリズムも考えていきたい。横ズレの測定では、カメラによって物体の膨れ方の規則性を研究していく必要がある。

また、VRにおいての問題点であるラグの解決には、スマートフォンやPCのスペックや、Wi-Fiの通信状況など、問題点は考えられる。この部分について、今後も検討を重ねたい。

6. 参考文献

「Processing 画像処理」

<<http://opencv.blog.jp/processing/使い方>>

「Unity 画像処理」

<<http://www.urablog.xyz/entry/2017/07/06/225444>>

「UDP通信」

<<https://qiita.com/nenjiru/items/8fa8dfb27f55c0205651>>

「マイコン漬け」

<<http://morokyuu.way-nifty.com/blog/2014/12/processingudp-8.html>>

7. 謝辞

今回の研究を行うにあたり、顧問の藤野先生には多大なご指導を賜りました。また、同研究会の先輩方にも多くの助言をいただきました。この場を借りて、深く御礼申し上げます。