

# 重力に平衡した自由浮遊バグによる気流動の可視化

## その9 類似度指標と解析時間の関係

正会員 ○森太郎\*1  
正会員 小西崇永\*2  
正会員 絵内正道\*1  
正会員 羽山広文\*1

大空間 実測 気球 自動追跡

### 1 研究背景・目的

アトリウムや体育館といった大空間では、居住域空調や自然換気の併用など緻密な空調計画が要求される。しかし現状では、大空間で気流分布を実使用時に調査することは困難である。そこで、一連の研究報告では大空間の気流動を簡便に可視化できる手法として、自由浮遊バグによる気流動の可視化法を提案してきた。

しかし既往の解析手法<sup>1)</sup>では、①ディスプレイ上での気球の位置決定を手作業に頼っている、解析作業に多くの時間がかかる、②空間内でのビデオカメラの位置を特定しなければならないので、設置に時間がかかると共に設置場所およびレンズの視野に恵まれた物件でなければ測定が困難、という問題があり、一連の報告の中で<sup>2),3)</sup>①の問題を解決すべく、バグの自動追跡手法の開発を行ってきた。本報では、解析速度と精度の向上を目的に、既往の文献の手法に改良を加えた結果を報告する。

### 2 自動追跡手法

これまでの解析結果を受け、解析速度と精度の向上を目指してアルゴリズムの改良を行った。

#### 2.1 解析フローの改良

今回、開発した解析手法のフローは、基本的な概念(初期データの登録、逐次検索)は前回と変わらないが、解析対象の画像や開発言語、類似度指標等に改良を加えた。

前出の計算方法では背景のある画像を直接取り扱ったため背景の変化によってバグを見失い、追跡が不可能になる現象が多く生じた。<sup>1)</sup>そのため今回は図1の様に、同一のカメラの設定で、バグがない画像を背景として登録し、バグが移動している画像から背景を減算(減算した後に絶対値をとり、増幅する)することによってバグを背景から浮かび上がらせ、その画像を解析対象の画像とした。このノイズは日射や照明の細かな変化やカメラの構造から生じてくるもので、このノイズとバグの画像を分離するためのアルゴリズムが必要になった。

#### 2.2 画像データの保管方法

画像データのメモリ上への保管方法は図2のような入れ子構造として画像解析の簡易化を図った。

##### ① Pictクラス

PictクラスはWINDOWS bitmapデータそのもので、BMPHEADER

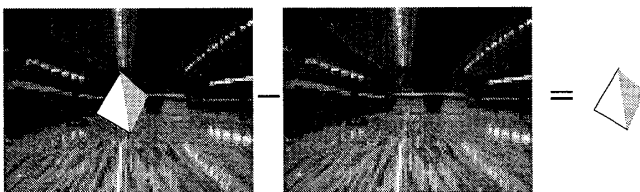


図1 バグ抽出の概念

クラス<sup>④</sup>(画像の大きさ等の情報の管理用)とPIXELクラス<sup>⑤</sup>(PIXEL情報の管理用)の集合から成る。本解析手法では減算を多用するため、式(4)、(5)の様にPictクラス同士の減算、代入演算ができるようにした。

$$\begin{aligned}
 PICT\_A &= PICT\_B - PICT\_C \\
 PICT\_A &= PICT\_B \quad (1) \\
 (PICT\_A &= PICT\_A - PICT\_B)
 \end{aligned}$$

$$PICT\_A = PICT\_B \quad (2)$$

##### ② BMPHEADERクラス

WINDOWS BITMAPの大きさや等の情報の管理を行う。

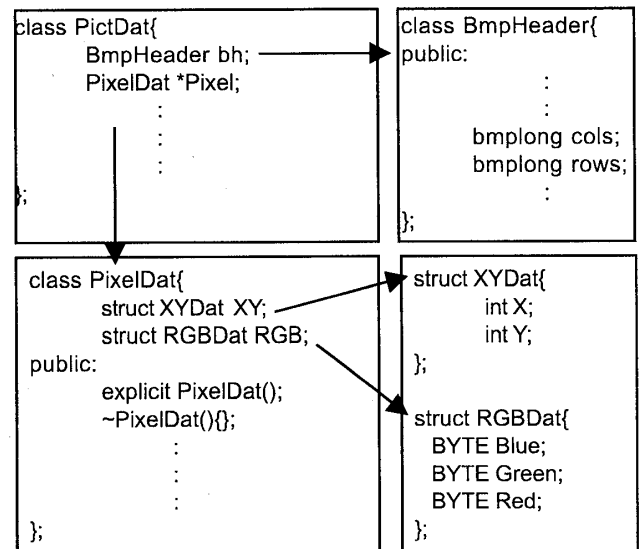
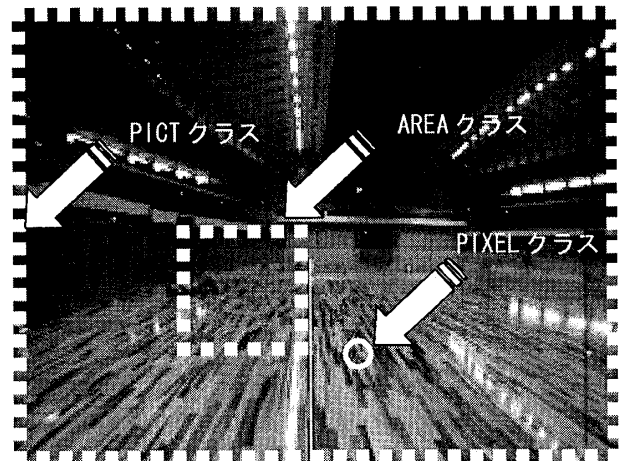


図2 画像データの保管方法

A visualization of air current by the use of balloons for gravity-free

MORI Taro, KONISHI Takahisa, ENAI Masamichi,

Part 9. The relation between the indexes of the analogy and the time for the analysis

HAYAMA Hirofumi

③PIXEL クラス

BITMAPを構成するPIXELデータの管理を行う。位置データ、XYDAT構造体(x:整数, x座標, y:整数, y座標)と輝度データ、RGBDAT構造体(RED:整数0-255, 赤色輝度データ, GREEN:整数0-255, 緑色輝度データ, BLUE:整数0-255, 青色輝度データ)から構成される。PICTクラスと同様にクラス同士の減算, 代入演算ができるようにした。

④AREA クラス

PICTクラスが保有しているPIXELクラスデータの集合の中から解析対象となる範囲(正方形)のPIXELデータを抽出した集合。PICTクラスと同様にクラス同士の減算, 代入演算ができるようにした。また解析対象のクラスとなるので類似度算出の元となるいくつかの指標をクラス内の関数として定義した。

2.3類似度指標

背景の減算によって、バグが暗闇の中に浮かんでいるような画像を追跡対象としているため、計算対象エリア内における分散をとることでノイズとバグを分離し、位置を確定する方法を採用した。

①中心規準の分散

先ず、エリア内の輝度データの二値化(閾値以下の輝度データを切り捨てる)を行う。その後、切り捨てられていない輝度データに関してエリアの中心を基準として分散をとる、それを類似度指標として利用する。

$$Resemblance = \sum(D^2)/N \quad (3)$$

D: 距離, N: 計算対象PIXELの数

②重みを付けた中心規準の分散

中心を基準とした分散をとった場合、エリアの辺縁に近い方が面積が多くなり、PIXELの持つ情報の密度が中心部と辺縁部では大きく異なってしまふ。エリア中心でバグをとらえるためには中心部の情報密度を高くした方がより効果的である。そこで(3)式のように母集団の数を距離で補正した。

$$Resemblance = \sum(D^2)/\sum(1/D) \quad (4)$$

D: 距離

③三点の分散

エリアの中心だけでなく、エリアの端点でも①と同様の分散をとる、それぞれの値を比較した。

3 解析結果

解析対象の画像群は、バグがカメラの奥行き方向に移動しているため、途中でバグの大きさが大きく変化する。初期のバグが大きいと解析領域を大きめに設定しなければならないため、小さくなったときにバグとAREA(解析対象領域)の大きさが合わなくなり、追跡が困難になる。従ってこのケースは最も追跡が難しいケースの一つである。

手動の結果と比較した解析結果と解析時間を図4に示す。手動の結果を正解とするならば、dt=0.2の場合も、dt=1.0の場合も大凡の動きを追跡することについては成功している。しかし、30枚目(dt=0.2)あたりから各指標とも手動の結果からは

ずれはじめている。これは自動追跡を行った結果、バグが解析対象領域の下側にきてしまっているのが分かる。このように大きさの変化への対応は今後考えていかなければならない課題の一つである。

また解析の速度に関しては、図4内に解析に要した時間を示してある。実際に解析に要する時間はI(画像のメモリ内への読み込みに要する時間)+II(自動追跡に要する時間)で表すことができる。解析時間とデータの分割時間(dt)との関係は、dtを長く(処理画像少)して検索エリア(SS)を大きくする場合とdtを短く(処理画像多)して検索エリア(SS)を小さくする場合があるが、表の結果からも分かるように、dtを短くし、検索エリアを小さくした方が解析に要する時間が短くなること分かる。これは、メインの計算部分のループがPIXEL毎の計算になっているため、dtは短くした分だけ一次関数の形で効くのにに対し、SSは二乗の形で効いてしまうためである。当然dtを非常に短くすれば精度が頭打ちになり、解析時間のみが増加する状況が出現するはずであり、この点は今後の課題とした。

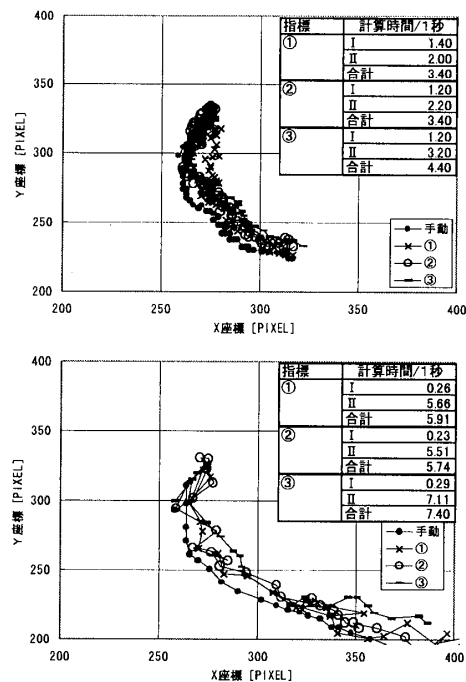


図4 解析結果, 類似度指標と解析時間の関係

【付記】

本研究は平成11~12年度文部科学省科学研究費補助金(萌芽的研究)の助成を得て行われた研究の一部である。

本研究はホクサイテック財団からの研究助成(平成11年度ホクサイテック財団研究開発支援事業, 若手研究者研究奨励補助金)を受けて実施された。

【参考文献】

- 1) 森太郎他: 重力に平衡した自由浮遊バグによる気流動の可視化, 日本建築学会技術報告集
- 2) 小西, 森, 絵内, 羽山: 重力に平衡した自由浮遊バグによる気流動の可視化 その6 バグの自動追尾手法の開発
- 3) 小西, 森, 絵内, 羽山: 重力に平衡した自由浮遊バグによる気流動の可視化 その7 バグの自動追尾手法の開発と適用

\*1 北海道大学大学院工学研究科, 博士(工)  
\*2 ㈱関電工  
\*3 北海道大学大学院工学研究科, 工学博士

\*1 Graduate School of Eng., Hokkaido Univ., Dr., Eng.  
\*2 Kandenko Co., Ltd.  
\*3 Graduate School of Eng., Hokkaido Univ., Dr., Eng.