

2010年度 修士論文

色情報を用いたサッカー映像における
選手の自動追跡

Automatic tracking of player positions from soccer
video image based on the color information

早稲田大学 大学院スポーツ科学研究科

スポーツ科学専攻 身体運動科学研究領域

5009A015-4

大坂 則之

Osaka, Noriyuki

研究指導教員： 誉田 雅彰 教授

目次

1.	緒言	1
1.1.	研究背景	1
1.2.	研究目的	2
2.	方法	4
2.1.	映像データ	4
2.1.1.	対象試合	4
2.1.2.	ビデオ撮影	4
2.1.3.	映像編集	5
2.1.4.	2台のカメラ映像の結合	5
2.1.4.1.	キャリブレーション	6
2.1.4.2.	ビデオ座標からフィールド座標への変換	9
2.1.4.3.	投影変換処理	10
2.1.4.4.	メディアンフィルタ処理	11
2.1.4.5.	色の調整	13
2.2.	色情報を用いた選手位置検出	14
2.2.1.	選手位置の自動検出方法	14
2.2.1.1.	カラー画像の二値化処理	14
2.2.1.2.	二値化画像の重心計算処理	16
2.2.2.	初期データの作成	17
2.3.	適応化処理	18
2.3.1.	検出枠位置の予測	18
2.3.2.	検出領域の制限	19
2.3.3.	二値化処理の閾値範囲と検出領域の拡大	20
2.4.	検出精度の評価方法	22
2.4.1.	正解データの作成	22
2.4.2.	選手位置の自動検出精度の評価	23
2.5.	追跡エラーの自動検出	24
2.5.1.	二値化画像のピクセル数に着目したエラー検出	24
2.5.2.	二方向カメラ映像を利用したエラー検出	24
2.5.3.	検出エラーの評価方法	26
3.	実験結果	28
3.1.	選手の追跡結果	28
3.2.	選手位置の自動検出精度の評価	28
3.3.	適応化処理を用いた自動検出結果	30
3.3.1.	検出枠の位置予測処理を用いた検出結果	30

3.3.2.	検出領域の制限処理を用いた検出結果	31
3.3.3.	二値化処理の閾値範囲および検出領域の拡大	32
3.4.	追跡エラーの自動検出結果	35
3.4.1.	二値化画像のピクセル数に着目した検出結果	35
3.4.2.	二方向のカメラ映像を用いた検出結果	36
4.	考察	40
5.	結論	43
	参考文献	45
	謝辞	46

1. 緒言

1.1. 研究背景

近年、情報機器の進展にともない、ビデオカメラは比較的安価で、小型で持ち運びも容易となった。さらに、ハイビジョンビデオカメラの登場により画像の解像度も向上し、長時間の撮影も可能になっている。スポーツ分野においても、ビデオカメラは多く使用されるようになり、撮影された映像は動作分析や戦略分析を目的として使用される場面が非常に多い。加えて、近年では、サッカーの試合中継で選手の走行距離や得点シーンなどで選手の移動軌跡が表示される等映像による分析は身近なものになってきている。

戦略分析の項目としては、フォーメーション、選手個人の移動パターンなどがあり、選手の移動パターンを把握するためには、ビデオ映像からフィールド上での選手の位置を抽出し、追跡する必要がある。この抽出結果は、試合中の移動距離や移動軌跡、あるいは、速度や運動量の変化を分析するために用いられる。選手の移動追跡に関する研究は広く行われており、特にサッカーに関するものが多く報告されている。例えば、分割テンプレートを用いた正規化相関法による神崎伸夫等の研究[1]、背景差分法を用いた時倉宋大等の研究[2]、色成分に着眼したレベルセット法を用いた林原局等の研究[3]がある。かつては、選手の位置情報の取得は手動で行われていたが、手動入力に要する手間を考えると、毎秒 30 コマで撮影した映像に対して選手 22 名を検出することになるので、1 秒当たりの選手位置の検出回数は 660 回となる。1 分当たりにすると 39,600 回、1 試合 90 分とすると 3,564,000 回となり、1 試合分のデータを取得するには、膨大な時間や労力がかかってしまう。そのため、実際の選手追跡では、人が監視しながら自動追跡を行い、追跡を誤った場合

には手動で検出をし直し、改めて自動追跡を再開するという方法が一般的に行われている。しかし、本来、分析を行う上で重要なことは得られたデータを基にして、次の試合へどう準備するかにか時間をかけるべきであるのに、現状ではデータを得る作業に膨大な時間がかかってしまう。

したがって、全自動あるいは極めて高い検出精度で選手追跡法を確立することは非常に意義があることである。全自動で選手追跡する方法を確立することで分析時間の短縮の端緒になると考え本研究を進めることにした。また、フィールド上の選手の位置データを取得することで、普段のトレーニングによるコンディショニングの成果が試合のパフォーマンスにどのような影響を及ぼしているかを把握することが可能となり、試合映像から特定のシーンを抜き出す映像検索にも有効に活用することができる。

このように、選手の評価や作戦分析、あるいは特定のプレイシーンを検索するための映像データベースの構築において、映像から自動的に選手の位置を検出処理は極めて重要な技術となる。

1.2. 研究目的

本研究では、色情報に基づくサッカー映像からの選手位置の自動追跡の方法を探る。

サッカーの試合映像で選手位置を検出する際、さまざまな問題が発生する。サッカーは、フィールド上で22人の選手がプレーし、接触を伴うスポーツであることから、ビデオ画像上で選手同士が重なり合い、追跡対象選手が隠れてしまうケース(オクルージョン)、屋外スポーツであるため、フィールド上の位置によって天候や照明などの影響を受けるケース等で追跡困難になる場合が多い。

そこで，本研究では，選手追跡の過程で起こる映像上の変化への適応化処理を用いることにより，追跡困難となる問題を解消する方法について検討を行う．特に，今回の検討では，検出枠の位置予測，選手同士の重なり合いへの対応，検出枠および二値化処理の閾値範囲の増減について検討し，より正確な選手位置の自動追跡法の確立を目指す．

2. 方法

2.1. 映像データ

2.1.1. 対象試合

本研究に使用したビデオ映像は、2009年6月28日に国立霞ヶ丘競技場で行われた第60回早慶サッカー定期戦の男子部の試合を用いた。試合は19時キックオフの45分ハーフで行われ、天候は雨であった。夜間の試合であったので、グラウンドには照明が当たっていた。

2.1.2. ビデオ撮影

サッカーのグラウンドは非常に広く、1台のカメラでグラウンド全体を撮影するのは困難である。また、1台のカメラをグラウンドから離れた箇所に設定して撮影した場合、ハイビジョンカメラを用いたとしても画像の解像度は不十分であり、ビデオ映像上で選手一人が占めるピクセル数が少なくなるため、選手位置の検出が難しくなる。

そこで、本研究では2台の固定カメラを用い、グラウンドの半面ずつを撮影することにした。撮影にはSONY社製のHDVビデオカメラを使用した。ビデオカメラはメインスタンドの最上階に設置し、それぞれのカメラでハーフコートのコーナー部分とハーフウェイラインが映るように配置した。図1に左右それぞれのビデオカメラからの撮影画像を示す。



左ビデオカメラ画像



右ビデオカメラ画像

図 1. 撮影画像

2.1.3. 映像編集

ビデオカメラで撮影した映像は、トムソン・カノーブス社の映像編集ソフト EDIUS NEO 2 を使用し、パソコンへの取り込みを行った。

ビデオ撮影時にはビデオカメラの同期処理を行わなかったため、2つの撮影映像に時間のズレが生じている。そのため、取り込んだ2つの映像中の選手やボールの動きから目視によって時間のズレをフレーム単位で決定し、時間のズレを補正することにより同期のとれた2つのビデオ映像を作成した。

各ビデオ映像は、ファイル形式を AVI ファイル形式に変換し、1試合分の撮影映像から3分間隔で切り出した映像を以降の処理で用いた。

2.1.4.2 台のカメラ映像の結合

サッカー映像から選手位置の追跡を行う処理において、2台のビデオカメラ映像を個別に用いると、選手が2つの映像間を行き来する場合に選手の検出処理が煩雑になってしまう。そこで、グラウンドの半面を映

した 2 つの映像から画像処理によって 1 つのフルコート映像を作成した。映像の作成は、プログラミング言語 MATLAB(2007b)で自作したプログラムを使用した。

2 台のカメラ映像の結合処理は、片方の映像(本研究では左カメラ)を基準画像とし、もう片方の映像(右カメラ)を基準画像上にはめ込むことによって行う。右画像の各ピクセルの座標値を投影変換を用いてフィールド座標値に変換し、次にこのフィールド座標値を逆投影変換を用いて左画像のビデオ座標値に変換することにより、右画像のビデオ座標値と右画像のビデオ座標値との関係が求められる。この関係式に基づいて、右画像の全てのピクセル値を左画像に置き換えることにより右画像の左画像へのはめ込みが行われる。以下、具体的なカメラ映像の結合処理について述べる。

2.1.4.1. キャリブレーション

キャリブレーションとはビデオ画像上の座標値(ビデオ座標)を実像の座標値(フィールド座標)に変換するための処理であり、実座標値の分かった被写体の画像を取り込み、コントロールポイントのフィールド座標値と画像上のビデオ座標値から、画像の座標変換式を求めるものである。今回の研究では、サッカープレイを 2 次元動作とみなし、選手の高さ方向は考えないこととする。したがって、選手のフィールド座標は 2 次元座標値とする。撮影した映像から、ビデオ座標上のコントロールポイント 4 点を選択し、DLT パラメータを決定する。

一般の 3 次元空間において、ビデオ座標系を $U \cdot V$, 対物面を $X \cdot Y$ 平面とすると、図 2 に示すように、デジタイズ面と $X \cdot Y$ 平面を平行にすることで、 $U \cdot V$ と $X \cdot Y \cdot Z$ の関係式が成り立つ。

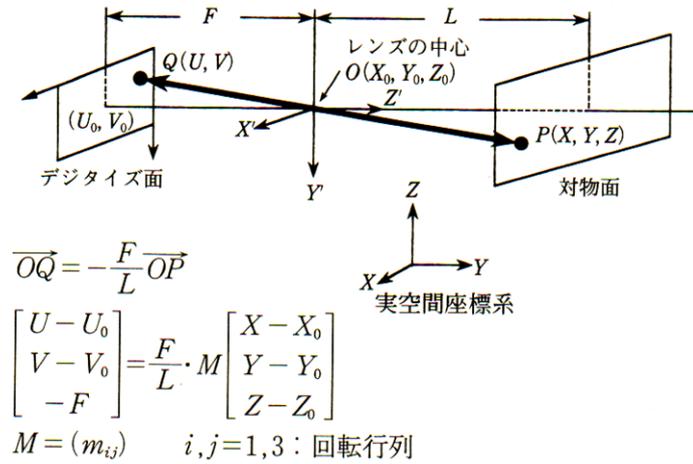


図 2. 一般的な実座標系とスクリーン座標系との関係

図 2 の M は 3×3 の行列である。

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix}$$

図 3 の式に M を代入すると、

$$\begin{bmatrix} U - U_0 \\ V - V_0 \\ -F \end{bmatrix} = -\frac{F}{L} \cdot \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix} \cdot \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix}$$

となる。これを行列計算すると、

$$\begin{aligned} U - U_0 &= -\frac{F}{L} [m_{11}(X - X_0) + m_{12}(Y - Y_0) + m_{13}(Z - Z_0)] \\ V - V_0 &= -\frac{F}{L} [m_{21}(X - X_0) + m_{22}(Y - Y_0) + m_{23}(Z - Z_0)] \\ -F &= -\frac{F}{L} [m_{31}(X - X_0) + m_{32}(Y - Y_0) + m_{33}(Z - Z_0)] \end{aligned}$$

となる。そして第 3 式で求まる L を第 1 式、第 2 式に代入すると、

$$U - U_0 = -F \frac{m_{11}(X - X_0) + m_{12}(Y - Y_0) + m_{13}(Z - Z_0)}{m_{31}(X - X_0) + m_{32}(Y - Y_0) + m_{33}(Z - Z_0)}$$

$$V - V_0 = -F \frac{m_{21}(X - X_0) + m_{22}(Y - Y_0) + m_{23}(Z - Z_0)}{m_{31}(X - X_0) + m_{32}(Y - Y_0) + m_{33}(Z - Z_0)}$$

$U_0, V_0, X_0, Y_0, Z_0, F$ は定数なので、定数項をまとめると、

$$U = \frac{L_1X + L_2Y + L_3Z + L_4}{L_9X + L_{10}Y + L_{11}Z + 1}$$

$$V = \frac{L_5X + L_6Y + L_7Z + L_8}{L_9X + L_{10}Y + L_{11}Z + 1}$$

となる。

本研究では 2次元位置に限定しているため、2次元位置のフィールド座標とビデオ座標の関係は 3次元の関係式から Z方向の要素を省くことにより次の式で表すことができる。フィールド座標 (X, Y) 、ビデオ座標 (U, V) とすると、

$$U = \frac{L_1X + L_2Y + L_3}{L_7X + L_8Y + 1}$$

$$V = \frac{L_4X + L_5Y + L_6}{L_7X + L_8Y + 1}$$

ここで $L_1 \sim L_8$ は DLT パラメータである。上式は、DLT パラメータを変数とする連立方程式になっている。

$$X_1L_1 + Y_1L_2 + L_3 - U X_1L_7 - U Y_1L_8 = U$$

$$X_1L_4 + Y_1L_5 + L_6 - V X_1L_7 - V Y_1L_8 = V$$

これで $L_1 \sim L_8$ の変数を決定する。 $L_1 \sim L_8$ を決定するには 4点以上のコントロールポイントを必要とする。 $L_1 \sim L_8$ を解くには次の 8元連立方程式を解くことで決定する。

$$\begin{pmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -U_1X_1 & -U_1Y_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -V_1X_1 & -V_1Y_1 \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -U_2X_2 & -U_2Y_2 \\ 0 & 0 & 0 & X_2 & Y_2 & 1 & -V_2X_2 & -V_2Y_2 \\ M & & & & M & & M & \\ M & & & & M & & M & \\ X_4 & Y_4 & 1 & 0 & 0 & 0 & -U_4X_4 & -U_4Y_4 \\ 0 & 0 & 0 & X_4 & Y_4 & 1 & -V_4X_4 & -V_4Y_4 \end{pmatrix} \begin{pmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \end{pmatrix} = \begin{pmatrix} U_1 \\ V_1 \\ U_2 \\ V_2 \\ M \\ M \\ U_4 \\ V_4 \end{pmatrix}$$

本研究では，図 3 に示したようにハーフコートの四隅の点をコントロールポイントに定め，左右それぞれの映像のキャリブレーションを行った．



図 3. 左ビデオ画像におけるコントロールポイントの設定

2.1.4.2. ビデオ座標からフィールド座標への変換

前項で求めた $L_1 \sim L_8$ を用い，ビデオ座標 (U, V) からフィールド座標 (X, Y) への変換を行う．1 台のカメラのビデオ座標値と 2 次元物理座標値の関係は，

$$\begin{bmatrix} L_1 - UL_7 & L_2 - UL_8 \\ L_5 - VL_7 & L_6 - VL_8 \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} U - L_3 \\ V - L_6 \end{bmatrix}$$

となり，これを計算すると，

$$\begin{cases} (L_1 - UL_7)X + (L_2 - UL_8)Y = U - L_3 \\ (L_5 - VL_7)X + (L_6 - VL_8)Y = V - L_6 \end{cases}$$

という二次元連立方程式となる．これを解くことでビデオ座標 (U, V) が決まるためフィールド座標 (X, Y) も決めることができるので，ビデオ座標からフィールド座標への変換が可能になる．

2.1.4.3. 投影変換処理

前項で述べた方法で，右画像のピクセルのビデオ座標をフィールド座標に変換する．次に右画像のピクセルのフィールド座標を左画像のビデオ座標に変換する．この作業を繰り返し行い，右画像の全てのピクセルに対し行うことで左右 2 つの画像を結合することができる．このとき，左画像のビデオ座標に右画像のどのピクセルも当てはまらないピクセルが生じる．このビデオ座標には黒のピクセルを当てはめた．図 4 に作成した結合画像を示した．

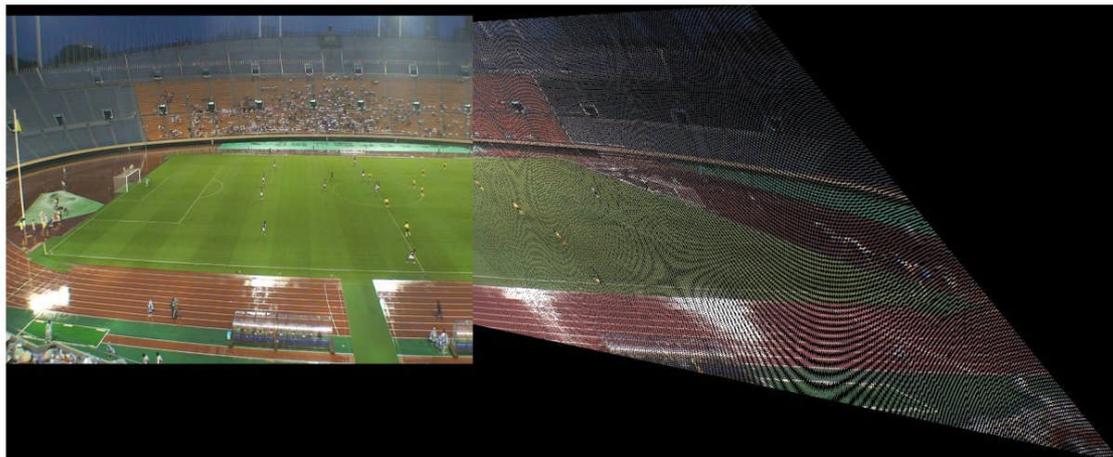


図 4. 投影変換処理による結合画像

2.1.4.4. メディアンフィルタ処理

投影変換処理を用い，画像の結合を行ったが，図 5 に示したように，結合部分には黒ピクセルによって補間されている．その黒ピクセルに対して，メディアンフィルタ処理を行い，補正を行った．

メディアンフィルタ処理とは，画像上のノイズを除去する手法である．図 6 に示すように，ノイズを周辺のピクセルの RGB 値の中央値で置き換えることで，ノイズを除去することができる．ただし，メディアンフィルタをかけると，画像の解像度は劣化してしまう．本研究では，画像の劣化を緩和させるため，黒(RGB 値が $R=0,G=0,B=0$)のピクセルの所のみメディアンフィルタ処理を行った．図 7 に，メディアンフィルタ処理後の画像を示す．

2.1.4.5. 色の調整

左右のカメラ映像には色の違いが観られたため、色の調整を行った。右画像の RGB 値を赤，緑，青のそれぞれの成分を 5 ずつ増減させ，左画像と見た目の色合いが同じになる値を求めた。今回は右画像の RGB 値の赤成分は変えず，緑成分に 10 加え，青成分から 20 引くことで左画像に近い色合いとなったので，この値を用い，色の調整を行った。

図 8 に，色調整を行った後の画像を示す。図に示すように，2 台のカメラ映像からグラウンド全体を映す良好な画質の映像が作成できていることが分かる。ただし，この画像結合法では，サッカー映像を 2 次元被写体として捉えているため，図 5 に示すように，変換された右画像上の選手の高さ方向の向きが斜めに傾いた画像となってしまう。これを解消するには被写体を 3 次元被写体として捉え，3 次元フィールド座標値を介して左画像と右画像の座標変換を決定する必要がある。その場合には，左右のグラウンド映像をそれぞれ 2 台のカメラで撮影し，3 次元の投影変換を用いて右画像から 3 次元フィールド座標値を求め，そのフィールド座標値から左カメラのビデオ座標値を求めることになる。

上記の画像結合処理を各フレーム画像について行い，グラウンド全体を捉えた映像を作成した。また，以降の選手位置検出処理では，その映像を 4 秒(120 フレーム)間隔で切り出し，試合開始から 2 分間分(3600 フレーム)の映像を使用した。



図 8.色の調整後の結合画像

2.2. 色情報を用いた選手位置検出

ここでは，サッカー映像から選手位置の自動検出法について述べる．
選手の位置検出は，以下の処理から構成される．

- (1) カラー画像の二値化処理
- (2) 二値化画像の重心計算処理
- (3) 検出枠の予測処理
- (4) 検出枠サイズおよび二値化閾値の適応化処理

以下，各処理を具体的に述べていく．

2.2.1. 選手位置の自動検出方法

2.2.1.1. カラー画像の二値化処理

選手追跡は，選手のユニフォームの色特徴に着目した手法を用い，選手 22 人全員を同時に行った．選手のシャツとパンツの色の組み合わせを特徴量とし，色情報に関する二値化処理と二値化画像における重心計算処理を用いて選手位置を自動検出した．

二値化処理は、あらかじめ指定したシャツとパンツの色の RGB 値を中心 に一定の範囲を設定し、画像のピクセル毎にピクセルの RGB 値が この範囲内にあればピクセル値を 1 とし、範囲外であれば 0 として二値 化画像を得た。ビデオ座標 i, j におけるピクセルの RGB 値を $(R_{i,j}, G_{i,j}, B_{i,j})$ と し、シャツの色の指定値を (R_s, G_s, B_s) 、パンツの色の指定値を (R_p, G_p, B_p) 、 色の閾値範囲を Δ 、二値化画像のピクセル値を $P_{i,j}$ とすると、二値化処理 は次式で表される。

$$if \begin{cases} R_s - \Delta \leq R_{i,j} \leq R_s + \Delta \\ G_s - \Delta \leq G_{i,j} \leq G_s + \Delta \\ B_s - \Delta \leq B_{i,j} \leq B_s + \Delta \\ R_p - \Delta \leq R_{i+h,j+h} \leq R_p + \Delta \\ G_p - \Delta \leq G_{i+h,j+h} \leq G_p + \Delta \\ B_p - \Delta \leq B_{i+h,j+h} \leq B_p + \Delta \end{cases} \quad then \quad P_{i,j} = 1, \quad else \quad P_{i,j} = 0$$

ここで、 h はシャツとパンツの相対的なピクセル間隔を表す。すなわち、ピクセルの RGB 値がシャツの色条件を満たし、かつその h ピクセル分下 および右のピクセルの RGB 値がパンツの色条件を満たす場合に、二値 化結果を 1 とし、それ以外の場合は 0 とする。シャツとパンツの色の組 み合わせを用いた二値化処理を行うことにより、両チームの選手を区別 できるようになり、両チームの選手が近付いた場合、あるいは画像中で 重なり合った場合でも、他チームの選手の占める二値化画像上のピクセ ル値は 0 となるため影響を受けなくなる。二値化処理では、指定される 色の RGB 値と色の範囲によってその結果が異なってくる。これらのパ ラメータ値は実験的に決定した。

図 9 に、検出枠内における選手の元画像と二値化画像を示す。

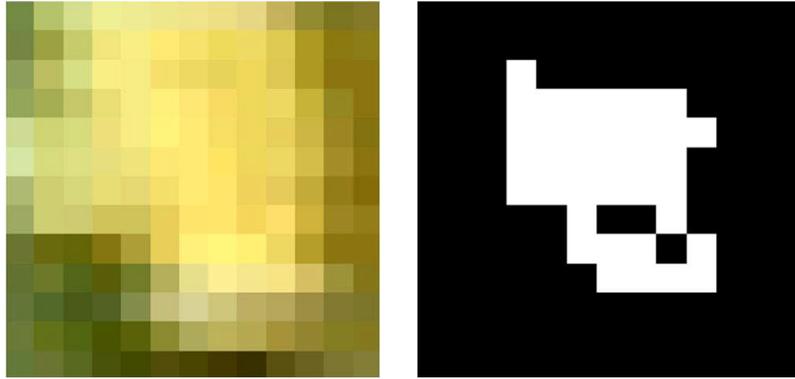


図 9.二値化処理の例

(左:検出枠内の元画像 右:二値化画像)

2.2.1.2. 二値化画像の重心計算処理

二値化画像の重心計算処理は，映像中の選手の検出領域をあらかじめ設定し，検出領域内での二値化画像に対して行った．検出領域内のピクセル座標を $n_1 \leq i \leq n_2$, $m_1 \leq j \leq m_2$ とし，二値化画像を P_{ij} とすると重心の座標値 (x_g, y_g) は次式によって計算される．

$$x_g = \frac{\sum_{i=n_1}^{n_2} \sum_{j=m_1}^{m_2} i p_{ij}}{\sum_{i=n_1}^{n_2} \sum_{j=m_1}^{m_2} p_{ij}}$$

$$y_g = \frac{\sum_{j=m_1}^{m_2} \sum_{i=n_1}^{n_2} j p_{ij}}{\sum_{i=n_1}^{n_2} \sum_{j=m_1}^{m_2} p_{ij}}$$

ただし，二値化画像が全て 0 のときは，検出枠の中心の座標値を重心の座標値とした．選手位置は上記重心計算により，得られるビデオ座標値として求めた．図 10 に，二値化画像の重心計算によって得られた選手位置を示す．図中の * 印が選手位置を示している．



図 10.重心計算による選手位置の決定

なお，閾値範囲や検出領域の大きさは，実験的に最適と思われるパラメータ値を設定した．図 11 に，22 人の選手位置の検出例を示す．図中の赤四角は各選手の検出枠を示し，検出枠内に+印は選手位置を示す．

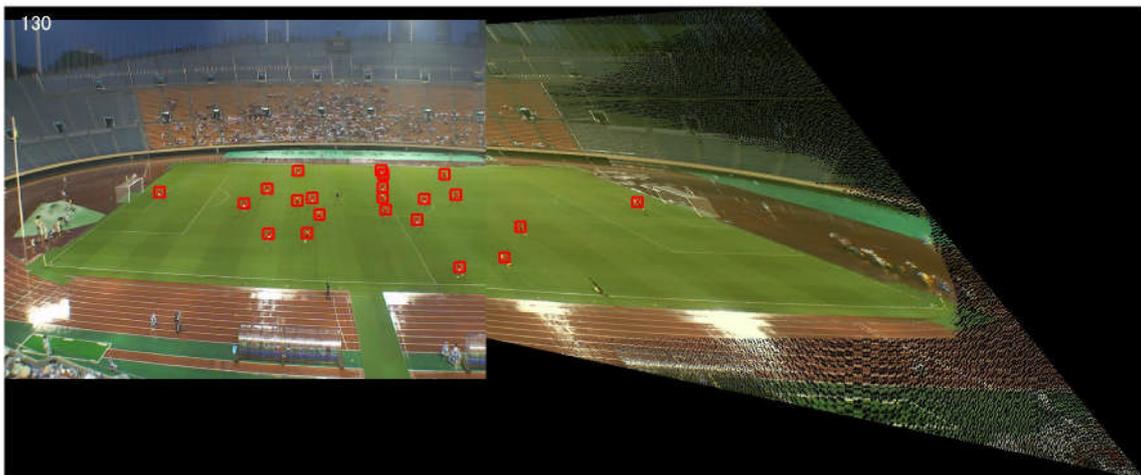


図 11.自動選手追跡

2.2.2.初期データの作成

上記で述べた選手位置の検出方法では，選手のシャツとパンツの色をあらかじめ指定しておく必要がある．そこで，試合開始時の選手位置と

検出対象とする色情報を取得し，選手 22 人分のテンプレートとなる初期データを作成した．ビデオ画像上でユニフォームのシャツとパンツの色の RGB 値を選手ごとに指定し，それらの相対的な位置と組み合わせを特徴量とした．なお，画像結合による右画像結合部の変形に対応するために，図 12 に示すように，左右の画像上での角度差を求め，回転させることにより，左右の画像ごとに相対位置を設定した．図中の赤丸は選手の初期位置とシャツの色の指定位置を示し，白丸はパンツの色の指定位置を示している．

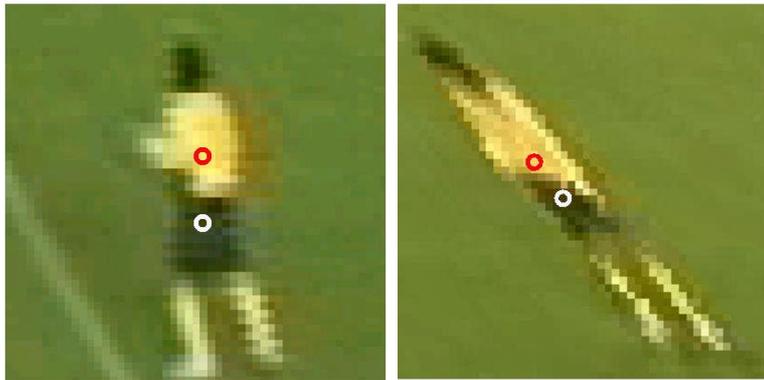


図 12.初期データ作成例

2.3. 適応化処理

2.3.1. 検出枠位置の予測

選手の移動スピードが速い場合には検出枠内に選手が入らなくなり，追跡困難となる．これを解決する方法として，検出枠の位置を予測する処理を用いた．検出位置の時間的な推移に応じ，適応的に検出枠を移動させる方法を用いた．あるフレーム $t(t \geq 2)$ における検出位置の座標値を (igx_t, igy_t) とすると，次フレームにおける検出枠の中心の座標値 (ix, iy) は次式によって計算される．

$$ix = igx_t + k(igx_t - igx_{t-1})$$

$$iy = igy_t + k(igy_t - igy_{t-1})$$

ここで、 k は予測係数であり、係数が 0 のときは検出位置を検出枠の中心とすることを意味する。予測係数のパラメータ値は実験的に決定した。図 13 に検出枠の位置の予測例を示した。図中の赤四角が前フレームの検出枠を示し、白四角が予測された次フレームの検出枠を示している。

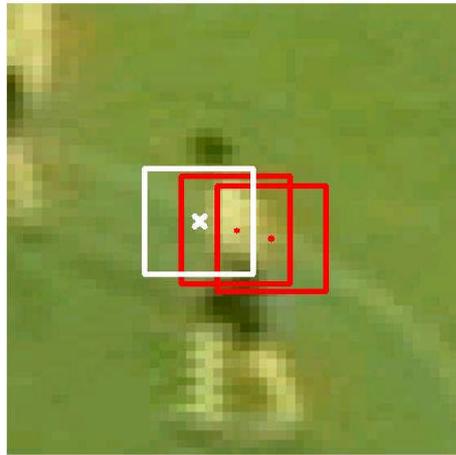


図 13.検出枠の位置の予測

2.3.2. 検出領域の制限

選手同士が映像上で重なり合うことによる選手位置の誤検出に対処するため、検出領域を制限する方法を導入した。選手同士がビデオ映像上で近付いたり、重なり合ったりした場合に同じ選手を検出しないように、一旦検出された選手の検出枠を他の選手の検出枠から除く処理を行った。すなわち、検出枠が重なりあった場合には、既に用いられた検出枠の領域では二値化処理におけるピクセル値を 0 とした。なお、制限する範囲は検出枠の大きさに係数を乗じた。係数のパラメータ値は実験的に決定した。図 14 に検出領域を制限した例を示した。図中の四角は 2 選手そ

それぞれの検出枠を示し，検出枠が重なり合った青で示された領域は検出枠から除かれる領域を示している．

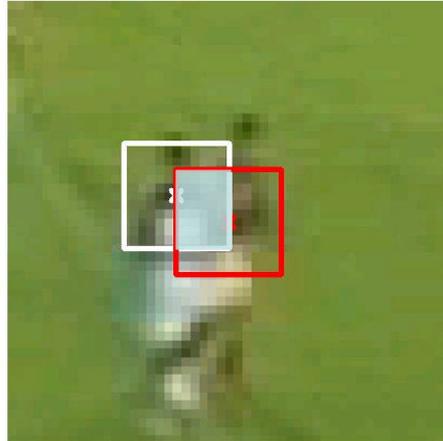


図 14.検出領域の制限

2.3.3.二値化処理の閾値範囲と検出領域の拡大

日光や照明の影響で，ビデオ画像上での RGB 値が変化し，時間の経過や選手の位置により追跡困難となるケースがあった．RGB 値の変化は 2.2.2 項で作成した初期データの色特徴と著しく異なる可能性があり，追跡対象選手であっても検出できなくなる．図 15 に時間経過による RGB 値の変化を示した．図中の赤線は二値化処理の閾値を示し，水色の線は誤検出と判定されたフレームを示している．

この RGB の変化による誤検出の問題を解決するために，二値化処理後のピクセル数に応じ，二値化処理の閾値範囲を拡大する処理，および，検出領域を拡大する処理を用いた．二値化処理後のピクセル数が設定した閾値を越えなければ，二値化処理の閾値範囲，および，検出領域を拡大する．図 16 に二値化処理の閾値範囲の拡大の例を示し，図 17 に検出枠の拡大縮小の例を示した．図 17 中の赤四角が現フレームの検出枠を

示し、白四角が拡大縮小後の検出枠を示している。

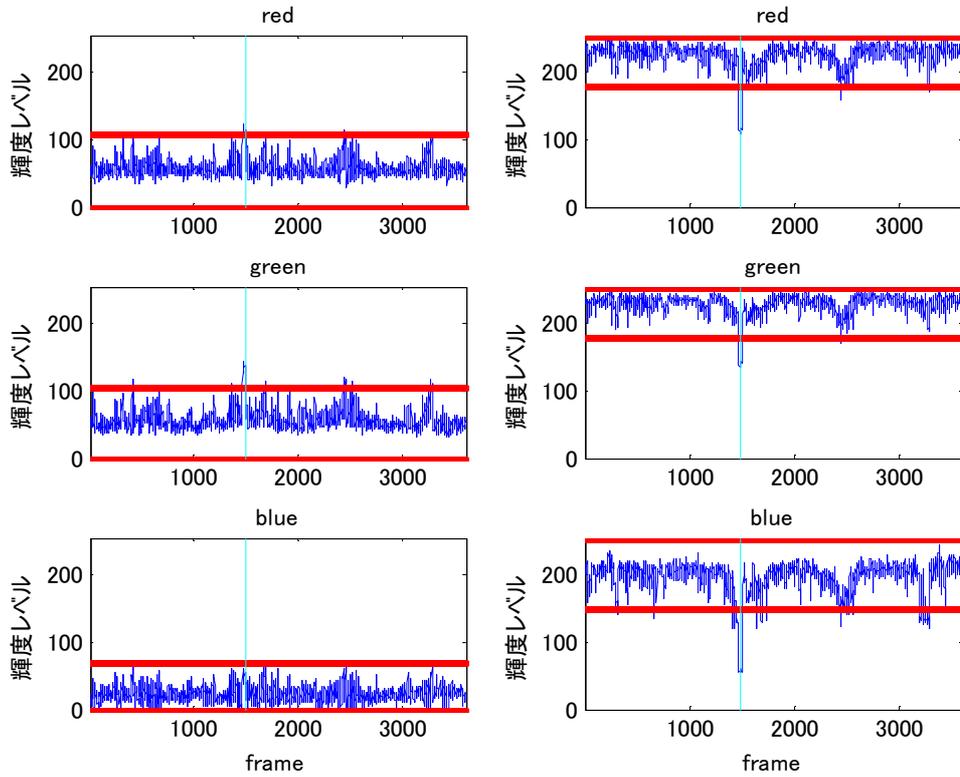


図 15.RGB 値の時間的变化

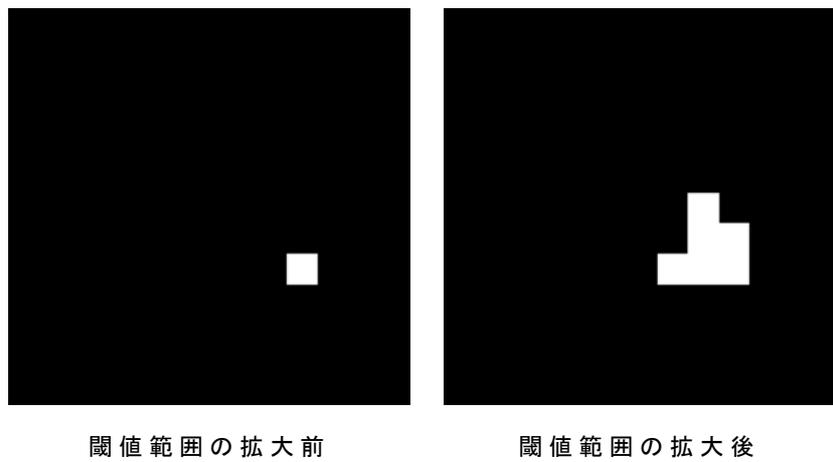


図 16.二値化処理の閾値範囲の拡大

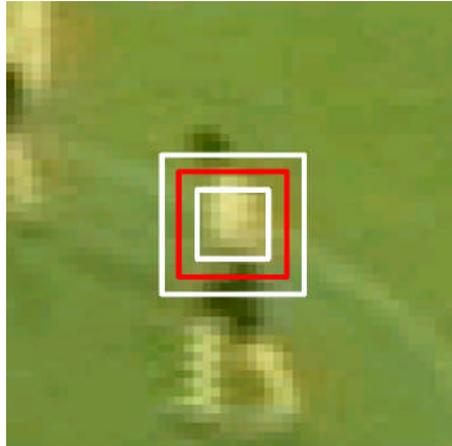


図 17.検出枠の拡大縮小

2.4. 検出精度の評価方法

上記で述べた選手位置検出法では，検出枠を検出された選手位置に応じて設定するため，一旦誤検出が生じるとその後正しい選手位置の追跡が困難になってしまう．したがって，誤検出が生じた以降のフレームは全て誤検出となり，適切な検出性能の把握ができなくなってしまう．そこで，検出処理において選手位置に大きな誤検出が生じた場合には，検出領域を正しい位置に戻して選手位置の自動追跡を継続し，選手位置の検出精度を評価した．

2.4.1. 正解データの作成

選手位置の自動検出精度の評価のために，半自動での選手位置検出処理を用い，事前に選手位置の正解データを作成した．正解データを作成には，プログラム言語 MATLAB(2007b)で作成したプログラムを使用した．図 18 に正解データ作成のプログラムの画面を示した．

正解データの作成手順は，2.2.2.項で作成した初期データを用い，対

対象選手を指定し、選手の自動追跡を開始する。選手追跡を監視しながら、対象となった選手が他の選手と映像上重なり合う等の問題により、指定した検出枠が対象選手から離れてしまい、自動追跡が困難になった場合は自動追跡を停止する。そして、問題になった場面に戻り、手動によるデジタイズを行い、選手位置を修正し、自動追跡を再開する。この作業を最終フレームまで繰り返し行う。

このようにして、全 22 選手の 3,600 フレーム分の選手位置の正解データを作成した。

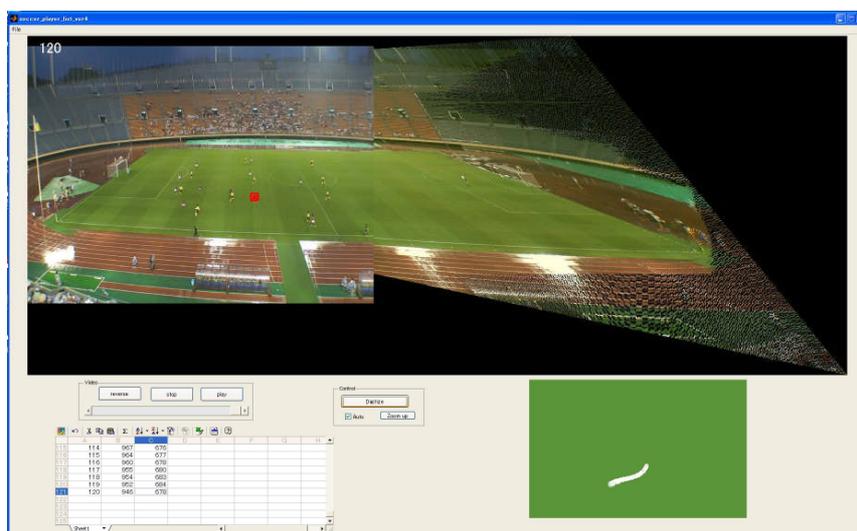


図 18.正解データ作成用のプログラムの画面

2.4.2.選手位置の自動検出精度の評価

自動検出によって取得した選手の位置データと 2.4.1.項で作成した選手位置の正解データを比較し、検出精度を評価した。取得データと正解データのそれぞれのフレーム総数は 79,200 である。取得データと正解データのビデオ画像上でのピクセルの距離を求め、あらかじめ決めた閾値を越えたとき、誤検出と判定した。本研究では、閾値を 30 ピクセル

と設定した。

2.5. 追跡エラーの自動検出

2.4.節でも述べたように，この選手位置の自動検出法では，一度でも選手の位置を見失うと正しい追跡ができないため，追跡ミスが生じたら，手動入力で選手位置を訂正し，自動追跡を継続する。しかし，追跡ミスを手動で訂正するためには，選手の検出状況を常時監視する必要があり，作業者を長時間拘束することとなる。また，作業者が追跡ミスを見過ぎさないという保障はない。そこで，追跡ミスを自動的に検出する方法を検討した。

2.5.1. 二値化画像のピクセル数に着目したエラー検出

2.2.1.項で述べたように，自動検出法では，カラーの二値化処理によって得られた二値化画像の重心計算により，選手位置を検出する。そのため，二値化画像のピクセル数が0であると，重心位置が求められず，選手位置を検出できない。すなわち，検出ミスが生じるときは二値化画像のピクセル数が0である状態が続くと考えられる。

本研究では，二値化画像のピクセル数が0である状態が5フレーム続いたら検出エラーと判定した。

2.5.2. 二方向カメラ映像を利用したエラー検出

これまで，一方向のカメラ映像での選手位置の自動検出法を検討してきた。自動検出に加え，誤検出時の手動訂正を併用する場合でも，高い検出精度と同時に選手追跡ミスの自動検出方法が求められる。このような自動検出処理における本質的問題点を緩和する方法として，カメラ方

向が異なる 2 台のビデオカメラを用いて撮影し、2 つのビデオ映像から求められる特定の選手のフィールド座標を元に選手位置の追跡ミスを検出することが考えられる。もし、2 台のビデオ映像から求められる選手のフィールド座標値が同一ならば選手位置が正しく検出されているとし、両者のフィールド座標位置の距離が一定以上離れる場合には、いずれか一方のビデオ映像上で検出された選手位置に誤りがあると考えられる。

このような方法により誤検出を自動的に見出すことができる根拠として、2 台のビデオ映像で同時に選手の誤検出が生じる確率は極めて小さいこと、誤検出は映像上で味方チーム同士の選手が重なり合う場合に生じやすいが、一方の映像では選手同士が重なり合っている場合でも、他方の映像では選手が離れていることが多いためである。選手位置の追跡ミスを自動的に検出することができれば、作業者は追跡処理結果を常時監視する必要がなくなり、誤検出が生じた時に作業者にそのことを知らせる仕組みを考えておけばよい。また、誤検出がどの選手を追跡中に生じたかも自動的に判断できるので、選手位置を手動で入力する際の手間も軽減される。

具体的な方法としては、ゴール裏の映像とメインスタンド側の映像から見える同一の追跡対象選手のビデオ座標を求め、それぞれのビデオ座標をフィールド座標に変換する。ゴール裏の映像から見える追跡対象選手のフィールド座標 FA 、メインスタンド側の映像から見える追跡対象選手のフィールド座標 FB とすると、 $FA=FB$ または $|FA-FB| < dth$ の場合は選手位置が正しく検出できたものとして自動追跡処理を継続し、 $|FA-FB| > dth$ の場合は追跡を停止し、手動入力により選手位置を指定する。このように自動処理と手動処理の両方を有効に使い分けることにより、正しい選手の位置を求めることができる。

ここで、閾値 dth の設定は注意を要する。 dth の値をより小さく設定するとより厳しく誤検出を判断することになり、検出ミスを見逃さない長所がある半面、誤検出でない場合にも検出誤りとみなすことが多くなり、手動入力の回数が多くなる。逆に、 dth の値を大きく設定すると、検出誤りをゆるく判断することになり、手動入力の回数は減るが、本当の誤検出を見逃すことにつながる。

上記の処理を行う上でもう一つ重要な問題がある。それは、ビデオ映像上の選手の重心位置は選手のほぼシャツの位置にあり、このビデオ座標からからフィールド座標に変換したのでは、正確なフィールド座標が求まらない。2次元フィールド平面上における正確な選手位置を求めるには、ビデオ映像上における選手の足元の座標値が指定されなければならない。そこで、重心位置と選手の足元の位置の間隔は一定ではなく、カメラに近い選手では遠い選手よりも間隔が大きくなる。そこで、検出された選手の Y 座標値と重心位置と選手の足元の位置の間隔との関係を回帰分析を用いて直線近似し、選手の Y 座標値から計算された間隔のピクセル数分 Y 座標値を修正することにより、選手の足元のビデオ座標値を求めた。

2.5.3. 検出エラーの評価方法

選手位置の正解データと検出位置との比較によって検知される誤検出のフレーム番号群 α と、FA と FB の距離の関係により検知される誤検出のフレーム番号群 β を比較することにより、誤検出の自動検出法を評価する。 α と β にあるフレームを比べて、一致する番号又は許容範囲内の番号差の番号のフレームは正解とする。 α にある番号で「 β にないかつ番号差が許容範囲内の番号が β にない」という条件を満たす番号のフ

フレームをミス誤りとする。Bにある番号で「 α にないかつ許容範囲内の番号差の番号が α にない」という条件を満たす番号のフレームを付加誤りとする。また、誤検出を正しく検出する正解率は、正解数とミス誤り数と付加誤り数の合計(総数)に対する正解数の割合、またミス誤り率と付加誤り率は、総数に対する各個数の割合として求めた。

3. 実験結果

3.1. 選手の追跡結果

選手の自動追跡を行った一例を図 19 に示す．図中におけるカラーの線は，自動検出された選手の移動軌跡を表している．

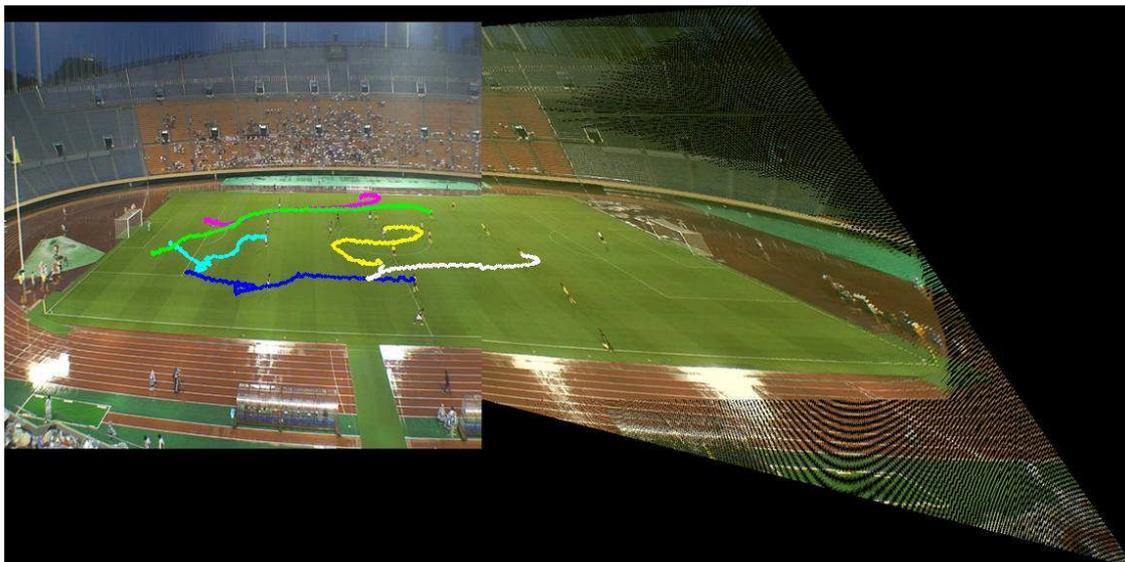


図 19.選手の移動軌跡の例

3.2. 選手位置の自動検出精度の評価

2.2.1 項で述べたように，二値化処理の閾値範囲および検出領域の大きさの設定は，色情報を用いる選手位置の自動検出法において検出結果に影響を及ぼす．二値化処理の閾値範囲が広い場合，二値化の条件を満たすピクセルが多くなり，重心計算の精度は高まるが，近づいてきた他の選手を誤って追跡してしまいやすくなる．逆に，閾値範囲が狭い場合，他の選手への誤追跡の可能性は小さいが，二値化の条件が厳しくなり，追跡対象選手であっても選手位置を検出できなくなりやすい．また，選手位置が移動した場合でも検出枠内から選手が外れることはなくなるが，

一方味方の複数の選手が検出枠に入り込む可能性が多くなり、その場合複数の選手に対する重心を求めることになり、誤検出につながる。逆に、検出枠を小さくすると、複数の選手が検出枠に入り込む可能性は小さくなるが、選手の動きによっては選手が検出枠内に入らず、選手位置を検出できなくなる場合がある。このように、二値化処理の閾値範囲および検出領域の大きさは相反する条件を満たす必要があり、これら2つは相互的に影響を及ぼすため、最適なパラメータ値を検討する必要がある。

図 20 および表 1 に二値化処理の閾値範囲と検出領域の大きさを変化させたときの選手検出正解率を示した。検出正解率が最も高かったのは、二値化処理の閾値範囲が 55、検出領域の大きさが 15 のときであり、検出正解率は 99.927%であった。検出領域の大きさごとにみると、検出正解率が二値化処理の閾値範囲が 50 から 65 の間でいずれも最高値を示した。

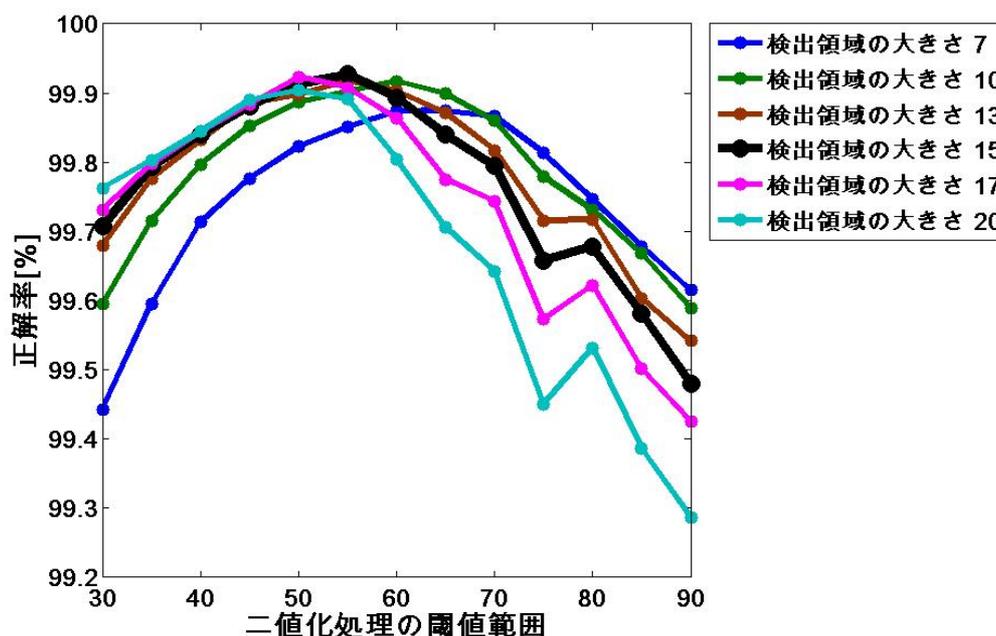


図 20.二値化処理の閾値範囲と検出領域と検出正解率の関係

表 1. 二値化処理の閾値範囲と検出領域の大きさの違いによる検出正解率

		二値化処理の閾値範囲												
		30	35	40	45	50	55	60	65	70	75	80	85	90
検出領域の大きさ	7	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.
		443	595	713	777	823	851	873	875	866	813	746	678	615
	10	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.
		595	716	797	852	886	902	917	899	860	779	731	668	590
	13	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.
		679	777	832	884	898	918	903	871	817	716	717	604	542
15	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	
	707	793	841	881	913	927	894	840	794	658	678	581	480	
17	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	
	731	798	845	884	923	908	864	775	744	573	623	501	426	
20	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	99.	
	763	803	845	890	904	891	804	706	643	451	532	386	287	

3.3. 適応化処理を用いた自動検出結果

3.3.1. 検出枠の位置予測処理を用いた検出結果

これまで、検出領域の中心位置は、前フレームで検出した選手位置を中心として移動させてきた。しかし、試合の状況により選手が全力で走る場面などでは、検出枠内から選手が外れてしまうことがある。2.3.1.項の方法で検出枠の予測処理を検討した。今実験では、検出領域の大きさと二値化処理の閾値範囲のパラメータ値は前節の結果を受け、検出領域の大きさを 15、閾値範囲を 55 と設定した。

図 21 に検出枠位置の予測処理を用いたときの選手検出正解率を示した。図中の横軸の値は予測係数であり，0 は検出枠の位置予測を用いていない場合と同じである。検出正解率が最も高かったのは，予測係数が 0.2 のときであり，検出正解率は 99.929%であった。予測係数が大きくなるほど，正解率が下がる傾向にあった。特に，1.2 を超えると正解率は急激に下がった。

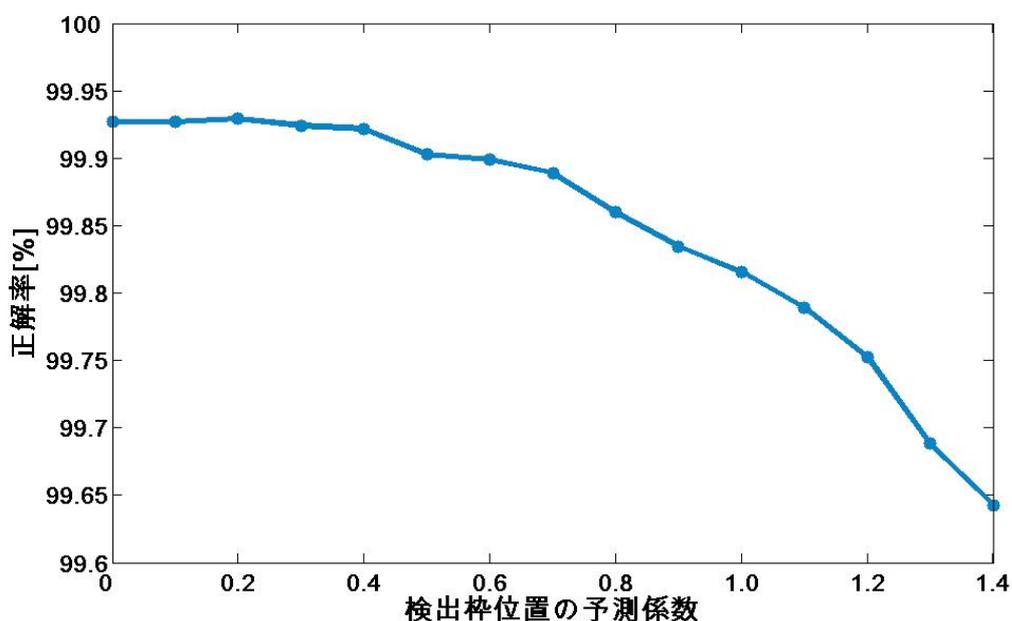


図 21.検出枠位置の予測係数と検出正解率の関係

3.3.2. 検出領域の制限処理を用いた検出結果

ユニフォームのシャツとパンツの色の組み合わせを用いた二値化処理を用いることで，他チームの選手が検出枠内に入ってきてても，選手位置検出に影響を受けなくなるが，一方，同チームの選手が検出枠内入ってくる場合，味方選手を誤って追跡してしまう可能性がある。そこで，2.3.2. 項の方法で検出領域を制限する処理を検討した。今実験では，検出領域

の大きさと二値化処理の閾値範囲のパラメータ値は前項と同条件とし、前項の結果を受け、検出枠位置の予測処理を係数 0.2 に設定して用いた。

図 22 に選手同士の映像上での重なり合いの対応とした検出領域の制限処理を用いたときの選手検出結果を示した。図中の横軸の値は制限係数であり、0 は検出領域の制限を行っていない場合と同じである。検出正解率が最も高かったのは、制限係数が 0.6 のときであり、検出正解率は 99.942% であった。制限係数が大きくなるほど、正解率が下がる傾向にあった。特に、1.2 を超えると正解率は急激に下がった。

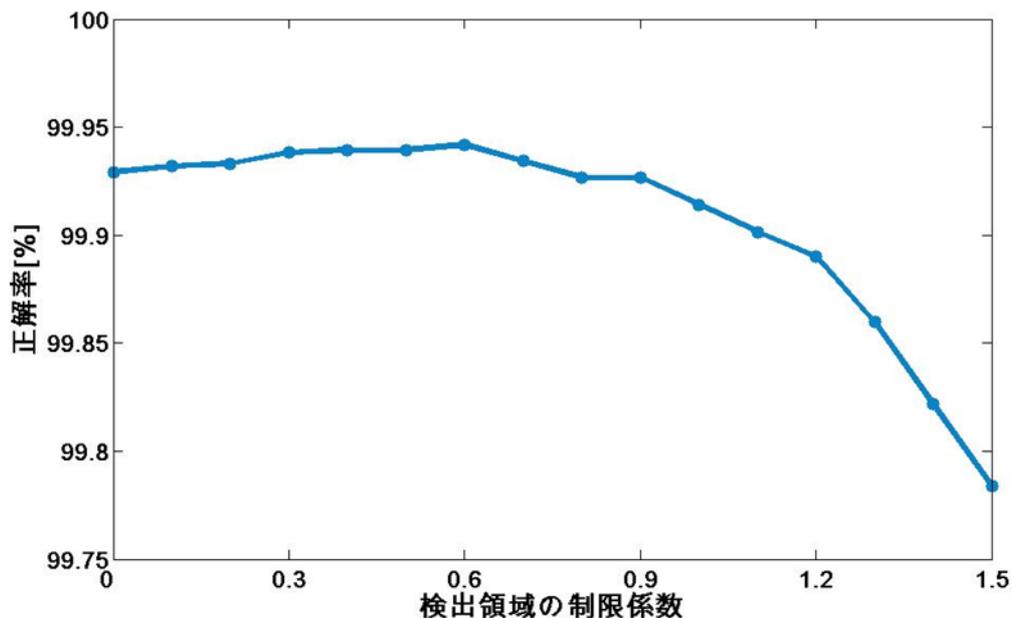


図 22.検出領域の制限係数と検出正解率の関係

3.3.3.二値化処理の閾値範囲および検出領域の拡大

2.3.3.項で述べたように、時間の経過とともに色の変化が見受けられ、対象選手であっても二値化処理の条件を満たせない可能性がある。その対応として、二値化処理の閾値範囲や検出領域を広げる処理を検討した。

今実験では、前項で用いた閾値範囲と検出領域のパラメータ値を最小値とし、適応化処理は前項の結果を受け、検出枠位置の予測処理に加え、検出領域の制限処理も用いた。制限係数は 0.6 とした。

図 23 に二値化処理の閾値範囲の最大値と検出領域の最大値を変化させたときの選手検出結果を示した。検出正解率が最も高かったのは、二値化処理の閾値範囲の最大値が 80、検出領域の最大値が 15 のときであり、検出正解率は 99.962%であった。全体的に閾値範囲の最大値が大きくなるにつれ、正解率が高くなることが分かる。

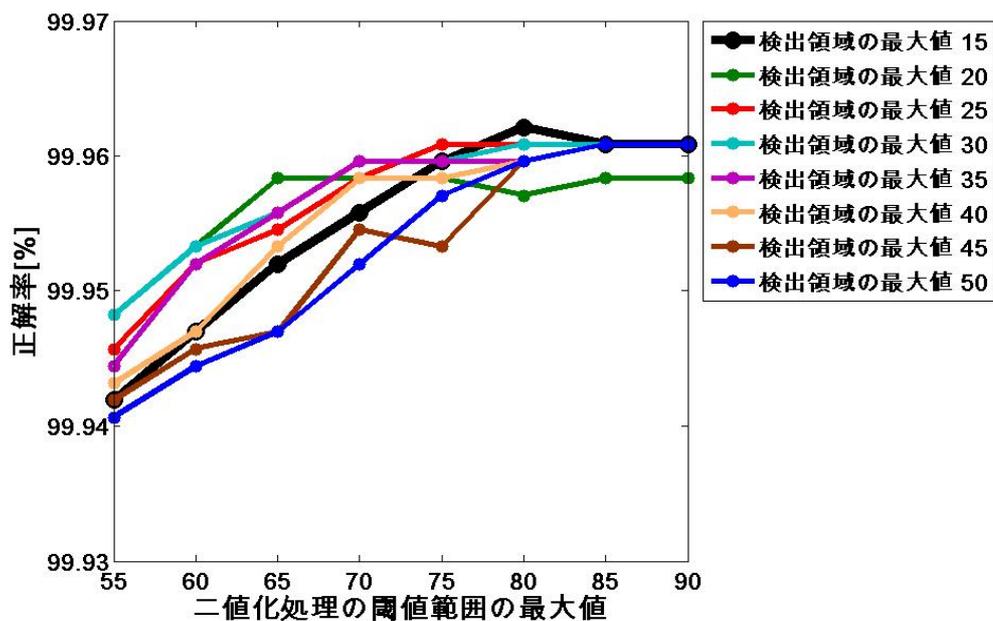


図 23. 二値化処理の閾値範囲の最大値と検出領域の最大値と検出正解率の関係

表 2.二値化処理の閾値範囲と検出領域の最大値の違いによる検出正解率

		二値化処理の閾値範囲の最大値							
		55	60	65	70	75	80	85	90
検出領域の最大値	15	99.942	99.947	99.952	99.956	99.960	99.962	99.961	99.961
	20	99.948	99.953	99.958	99.958	99.958	99.957	99.958	99.958
	25	99.946	99.952	99.955	99.958	99.961	99.961	99.961	99.961
	30	99.948	99.953	99.956	99.960	99.960	99.961	99.961	99.961
	35	99.944	99.952	99.956	99.960	99.960	99.960	99.961	99.961
	40	99.943	99.947	99.953	99.958	99.958	99.960	99.961	99.961
	45	99.942	99.946	99.947	99.955	99.953	99.960	99.961	99.961
	50	99.941	99.944	99.947	99.952	99.957	99.960	99.961	99.961

次に、図 24 に二値化処理の閾値範囲の増減値を変化させたときの選手検出結果を示した。図の横軸は前フレームの二値化画像のピクセル数に応じ、閾値範囲を拡大させる増減値である。検出正解率が最も高かったのは、増減値が 15 のときであり、検出正解率は 99.963%であった。