

課題部門
登録番号：10001

OMIAI

OMIAI（オミアイ）はバレーボールで”お見合い”をなくすシステムです。
お見合い範囲を可視化し、選手の配置をオンラインのみんな
考えて「成長」させることで守備力の向上を目指します。

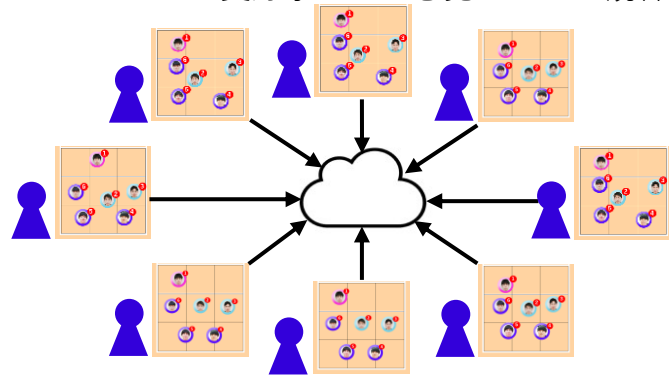
対象者
オンラインで参加する
全国のバレーボール
愛好家・選手・監督

はじめに

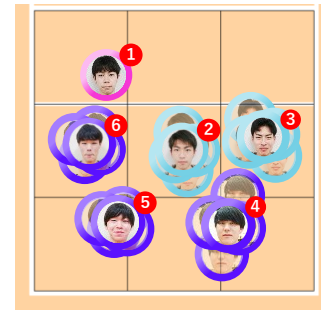
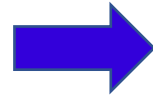
「バレーボールにおける最強の守備配置を知りたい！」

バレーボールの理想的な守備配置は見る人によって様々な形に変わります。

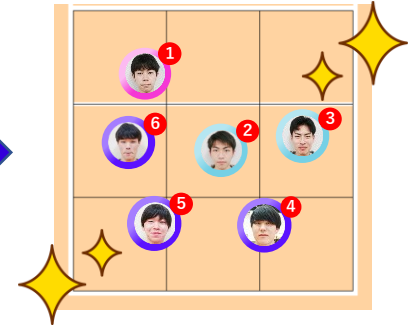
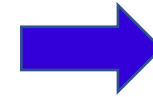
そこで、**スポーツ観戦をしながら周囲の人と理想の配置について語り合う体験**をインターネットで集約すれば
全国のバレーボール愛好家たちの意見により「観客が思う最強の配置」を**“集合知”**として生み出すことができるのではないかと考えました。



愛好家同士で理想の配置についての語り合い



みんなの意見を集約



集合知としての配置

私たちが目指す最強の守備配置は、バレーボールで一番もったいないミスである**“お見合い”**が起きない配置です。

守備範囲は目に見えないから
わかりづらい...

“お見合い”とは選手の間
にボールがきたとき、
ゆずり合ってしまう、
ボールを落としてしまう
ことをいいます。

お見合いが起きると...



相手チームに点を取られてしまう

お見合いが起きない配置だと...



確実にレシーブできる!

自分がレシーブ
すればいいんだ!

“お見合い”が起きない守備配置では、
だれがレシーブすべきかという**判断の遅れがなくなる**ので、
選手が動きやすくなります!

そこで、お見合い範囲を可視化し、最強を目指して**守備配置をオンラインで成長させる**システム

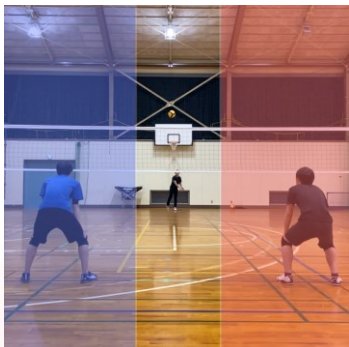
DMIAIを提案します!

システム概要

DMIAIはお見合い範囲を可視化し、選手の配置をみんなで考えるシステムです。
本システムでは、コートに立っている人を選手、監督も含めた選手以外の人を観客と呼称します。

問題点

お見合いは主に**選手同士のコミュニケーション不足**や**誰が取るべきなのかを理解できていない**ことによって発生します。



選手それぞれが**自分が取る**と思っている範囲



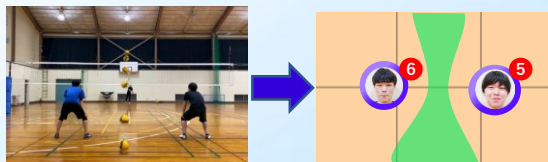
選手が**お互いに相手が取るだろう**と思っている**範囲**がお見合い範囲



守備配置をみんなで考えることでコミュニケーション不足を解消し、守備に対する理解を深めます。

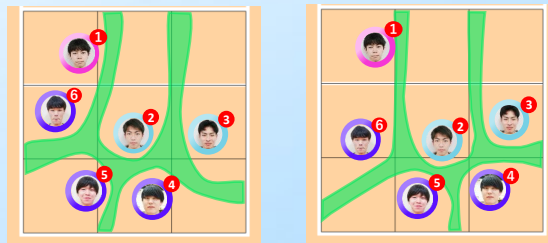
■お見合い範囲の可視化

オンラインで動画を公開し、客観的データを収集します。客観的データとあらかじめ取得してあった主観的データをもとに機械学習でお見合い範囲を可視化します。



■守備配置のシミュレーション

選手と観客には、選手コマの配置をしながらお見合い範囲の少ない配置を目指してもらいます。お見合い範囲は選手コマを配置し直すたびに再計算されます。



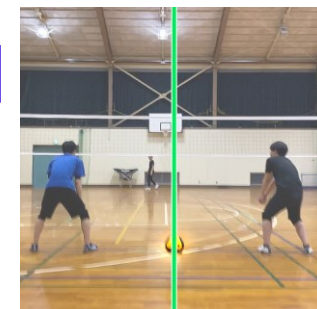
本システムでは選手が判断する**主観的データ**と、観客が判断する**客観的データ**を利用します。

主観的データ



選手の判断では、右の選手の方がレシーブが得意な場合、このボールは右の選手が取るのが最適だという意見になります。

客観的データ



観客の判断では、2人の真ん中に落ちるボールは、どちらの選手がとるべきかの意見が**1:1**に分かれる可能性が高いです。

主観的データでは選手同士の心理関係が加味されます。心理関係とは自分より相手の方がレシーブがうまいと思っているかどうかなど、選手の判断に作用する心理的な関係のことをいいます。

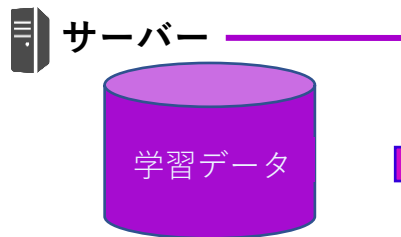
客観的データでは選手とボールの位置関係だけで、どちらの選手がボールを取るべきかを判断します。

システム構成

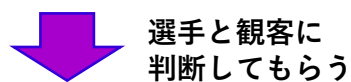
お見合い範囲作成API用の学習データ取得



誰がとるべきかを判断するための動画を撮影します。ボールが地面についた瞬間のボール座標を物体検出ライブラリを使って検出します。



選手・観客から集めた、誰がとるべきかの判断結果、ボールの着地点の座標、人物の位置座標を学習データとして取得します。



動画撮影する際に選手2人がボタンを持ちます。自分がとるべきと思ったらボタンを押してもらい、意見を収集します。この意見は選手のお見合い範囲を決める**主観的**データとなります。



動画を見て、どちらの選手がとるべきかを判断します。この意見は、選手のお見合い範囲を決める**客観的**データとなります。

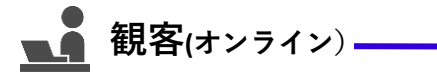
お見合い範囲をあらわす集合知の成長



学習データをもとに、Webの配置シュミレーションに対して、誰がとるべきかのクラスタリングを行います。



実際に集合知の配置図通りに選手がフォーメーションを組み、サーブレシーブをします。配置に対しての評価を返します。



選手の配置図



お見合い範囲

お見合い範囲がなるべく少なくなるように選手を動かして最適な配置を作ります。



選手の配置図



お気に入り度

配置データから経験年数別、世代別に集合知を表示させます。また、自分の作った配置と比較することができます。

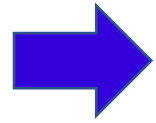
システムの流れ① 観客からのデータの収集

選手が並んでいる動画を見てどちらがボールを取るべきかを観客に解答してもらいます。

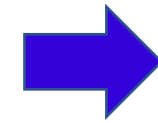
Webページ



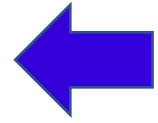
自分のバレー経験年数を選択します。バレー経験年数は配置の集合知を出すとき、経験年数別に表示するために使います。



選手が並んでいる状態で、サーブが飛んでくる動画を流します。選手とボールの位置関係をわかりやすく示すために、選手はレシーブせず立っているだけです。



違う動画で
繰り返します。



ボールが着地した時点で、どちらの選手がボールを取るべきかを判断してもらいます。解答後、他の人が選んだ割合を表示します。自分の意見は多数派か少数派か確認することができます。



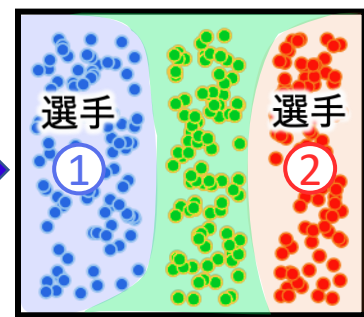
データの収集

主観的データ

左表は主観的データをお見合い範囲に変換する判断基準です。主観的データはシステム概要スライドで説明した、選手本人から取得するデータです。どちらの選手がとるべきか判断がそろった場合は、その選手の色ラベル、判断が合わずお見合いになる場合は緑ラベルとします。

選手	ラベル
① ②	
自 自	緑
自 相	青
相 自	赤
相 相	緑

コートの平面座標



主観的・客観的データをそれぞれプロットすると、このように緑でお見合い範囲がわかります。

客観的データ

左表は客観的データをお見合い範囲に変換する判断基準です。どちらの選手がとるべきかで、片方の選手に60%以上の判断がされた場合はその選手の色ラベル、判断が合わずどちらの選手も60%以上の判断がされなかったら緑ラベルとします。

選手	ラベル
① ②	
40~60%	緑
60%以上	青
60%以上	赤

システムの流れ② 配置シミュレーション

画面内のコート上に可視化されたお見合い範囲を参考に、最適な守備配置を考えます。

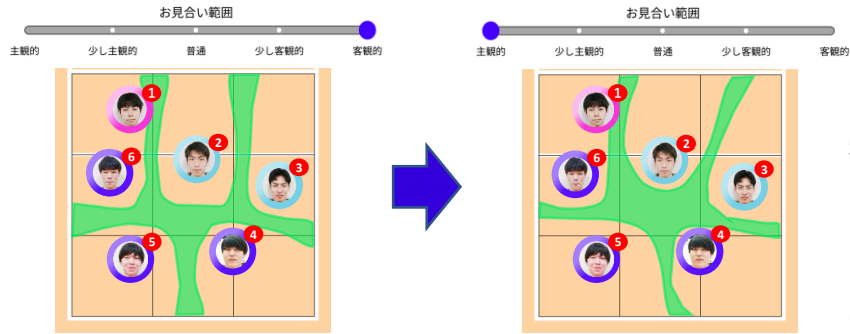


自分のバレー経験年数を選択します。



選手6人とお見合い範囲が表示されます。選手を動かして最強の配置を作ってください。

お見合い範囲の学習データを変える

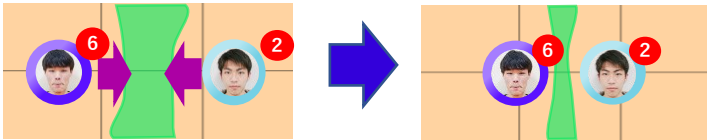


お見合い範囲の計算に使う主観的・客観的データの割合をスライダーで変化させます。

主観的・客観的データでどれほどの違いがあるかを見比べることができます。

配置の基本

①選手を動かすとお見合い範囲が変わる



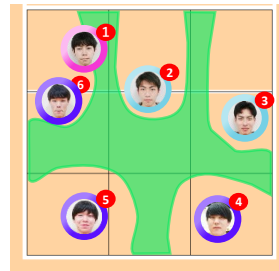
②セッターは所定の位置に固定されている

セッターはアタッカーが攻撃するためのトスを上げる必要があるため、一般的にサーブレシーブはしません。そのためお見合い範囲の計算には含めません。

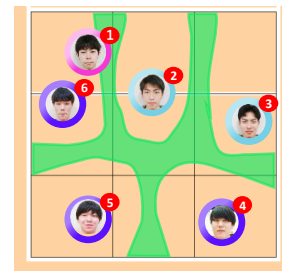
③各ローテーションごとの配置を考える

バレーはサーブ権をとると選手は時計回りに1つずつポジションを変更します。選手が一巡するまで計6パターンの配置を作ってください。お見合い範囲の少ない配置を目指しましょう。

コート上のお見合い範囲の割合を表示



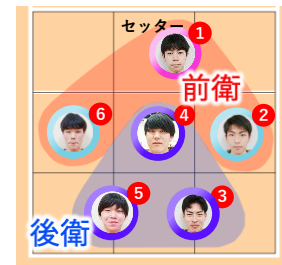
お見合い範囲：60%



お見合い範囲：40%

コートに対するお見合い範囲をパーセントで表示させます。選手コマを動かすたびに数値は変化します。パーセントが少ないほど、お見合い範囲が少なく、良い守備配置だといえます。

ローテーションする



ローテーションボタンを押すと、選手の配置がローテーションします。

セッターが前衛にいる場合はネット前で固定、後衛にいる場合はすぐにネット前にいけるように選手の後ろに固定とします。

システムの流れ③ 選手配置の閲覧

「集合知として成長」した選手配置を閲覧でき、自分の選手配置と比較することができます。



集合知変化バー

経験年数選択バー、世代選択バーを動かして集合知の表示を切り替えることができます。

自分の配置

配置シミュレーションで作った自分の配置を表示します。また、配置の情報としてお見合い範囲の割合を見ることができます。

集合知の配置

集合知の配置を見ることができます。また、配置の情報としてお見合い範囲の割合、お気に入り度を見ることができます。

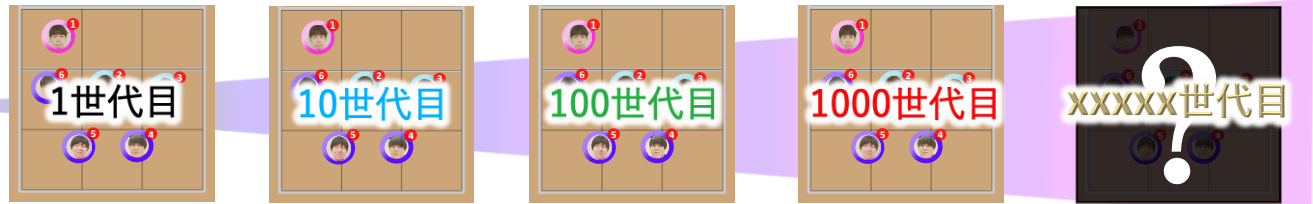
集合知の経験年数別表示

集合知を経験年数別で表示します。経験年数とお見合い範囲は相関があるでしょうか。



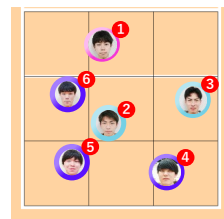
集合知の世代別表示

集合知を世代別で表示します。世代の進行による配置の変化を知ることができます。



選手の評価

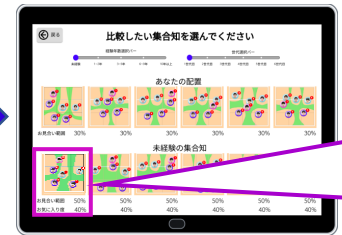
集合知配置を選手に実践してもらい、評価してもらいます。



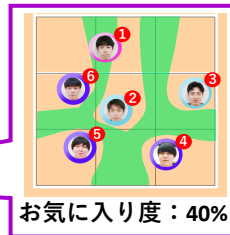
集合知の配置を選手が確認します。



集合知の配置通りに選手がフォーメーションを組み、サーブレシーブをします。



Webページで配置に対する選手の評価をお気に入り度として表示させます。

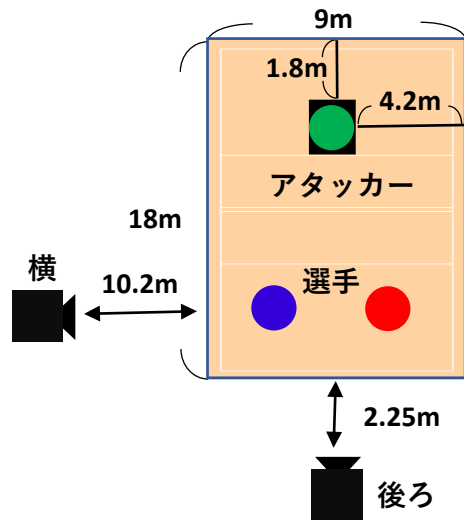


実現方法① 学習データ取得用動画の撮影・座標変換

学習データ取得用の動画を撮影し、スクリーン座標からワールド座標へ変換します。

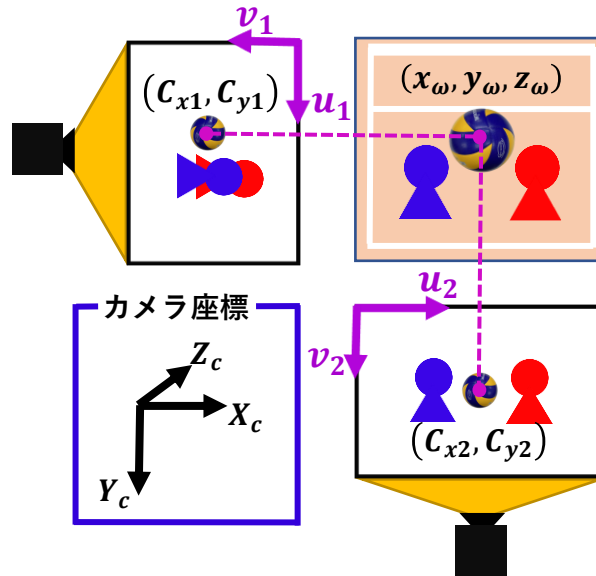
動画撮影

カメラ2台を選手2人の横、後ろに置きます。選手2人はボタンをもって指定の位置に立ってもらいます。アタッカーは台の上に乗って、ランダムにサーブを打ってもらいます。選手2人は前後、左右の位置を変えつつ動画を撮ります。

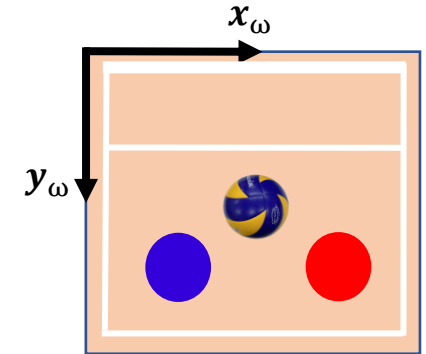


座標変換

コートを上から見たときのボール座標を取得するために、2台のカメラで撮った画像から変換式を連立させ、スクリーン座標からワールド座標へ変換します。



ワールド座標 (x_ω, y_ω) をコートを上から見たときのボール座標として使用します。



コートを上から見た図

(u, v) :スクリーン座標
 $(x_\omega, y_\omega, z_\omega)$:ワールド座標P点
 (C_x, C_y) :スクリーン上の焦点中心
A:カメラ内部パラメータ
R:回転行列
t:移動ベクタ

俺じゃない

俺がとる



選手には自分がとるべきだと思ったタイミングでボタンを押してもらいます。

スクリーン座標からワールド座標の変換式

$$Z_c \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = A(R|t)p = \begin{pmatrix} F_x & 0 & C_x \\ 0 & F_y & C_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix} \begin{pmatrix} x_\omega \\ y_\omega \\ z_\omega \\ 1 \end{pmatrix}$$

実現方法② お見合い範囲の計算

学習データをもとに、選手の配置画面にお見合い範囲を表示させます。

ロジスティック回帰による
クラスタリング

誰がとるべきかのラベル
選手①の (x, y)
選手②の (x, y)
ボールの落下点の (x, y)



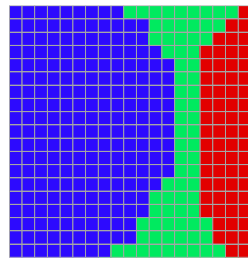
特徴量の例

選手同士のx差分
選手同士のy差分
選手①とボールのx差分
選手①とボールのy差分
選手②とボールのx差分
選手②とボールのy差分

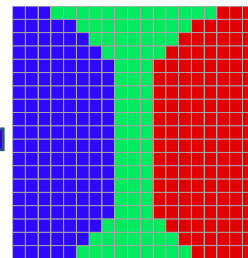
上図のデータから、学習データとなる特徴量を作ります。

特徴量を使って機械学習を行います。主成分分析、標準化、ロジスティック回帰を行い、お見合い範囲を求めるための式を主観・客観でそれぞれ作成します。

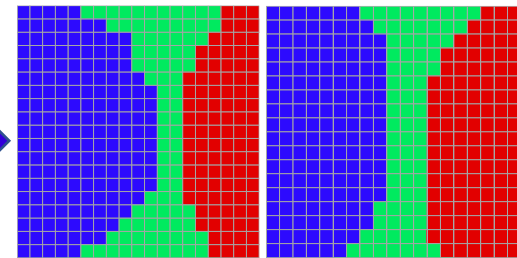
お見合い範囲作成API



主観的データのプロット



客観的データのプロット

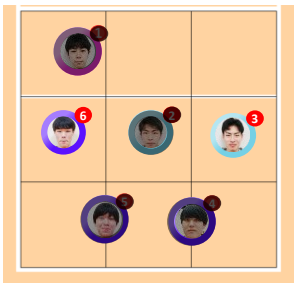


主0.75 : 客0.25の例 主0.25 : 客0.75の例

Webから取得した選手の座標をテストデータとして、機械学習から導出した式に代入します。ボールが落下する座標をコート平面のメッシュグリッドとして、誰がとるべきかを各座標ごとに判断します。

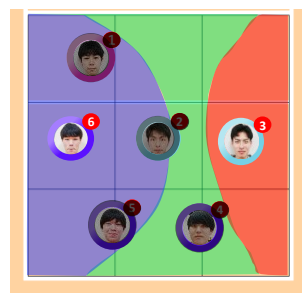
Webから取得したお見合い範囲割合変化バーの値に応じて、主観的・客観的データの重みを変えて合成します。その結果をWebに返します。

Webページ (配置画面)

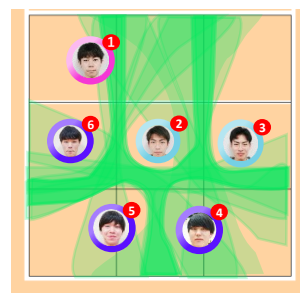


全員分の組み合わせが終わるまで繰り返します。

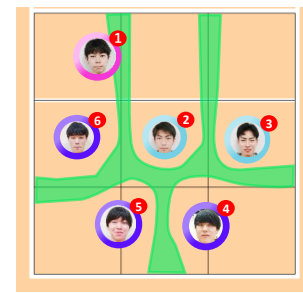
選手2人の(x,y)座標を取得し、お見合い範囲作成APIに送ります。



選手2人によるお見合い範囲をプロットします。



全ての組み合わせのお見合い範囲を重ねます。



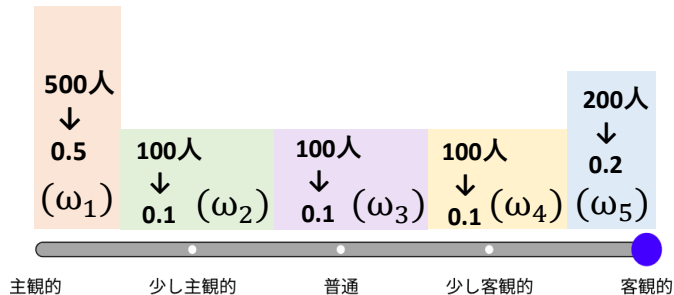
お見合い範囲に重みを付けて、一定の重みの割合で重なっている範囲のみを表示させます。

実現方法③ 集合知の算出

システム利用者が考えた各選手の配置座標 (x, y) の加重平均を取り、集合知を算出します。

集合知としての選手の配置を決める

観客がどういう割合で主観・客観を選んだかの回数を重みとします。



$$\bar{p}_\omega = \frac{\omega_1 p_1 + \omega_2 p_2 + \omega_3 p_3 + \omega_4 p_4 + \omega_5 p_5}{\omega_1 + \omega_2 + \omega_3 + \omega_4 + \omega_5}$$

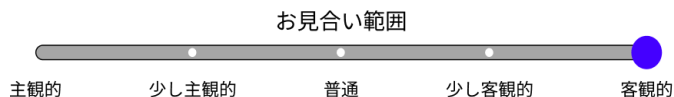
配置画面で選んだお見合い範囲割合変化バーの項目別に各選手の配置 (x, y) の平均値 p をとります。左の項目の平均値から順に p_1, p_2, p_3, p_4, p_5 とします。

お見合い範囲割合変化バーの項目別に重みをつけます。重みは項目を選んだ人数別で出します。左の項目の重みから順に $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$ とします。

左図の式に代入して選手の配置データの加重平均を出して、集合知として表示させます。

p : 配置データ (x, y) の平均
 ω : データの重み

集合知としての学習データの割合を決める

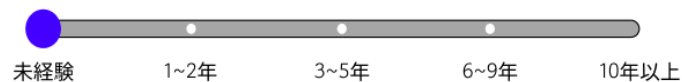


$$\bar{d}_\omega = \frac{\omega_1 d_1 + \omega_2 d_2 + \omega_3 d_3 + \omega_4 d_4 + \omega_5 d_5}{\omega_1 + \omega_2 + \omega_3 + \omega_4 + \omega_5}$$

お見合い範囲割合変化バーの項目別に、主観・客観データを混ぜる割合を変えたお見合い範囲を作り、重み付けで合成します。また、合成したお見合い範囲を選んだ人数別に重みを取ります。左図の式に代入してお見合い範囲の加重平均を出して、集合知の配置図のお見合い範囲として表示させます。

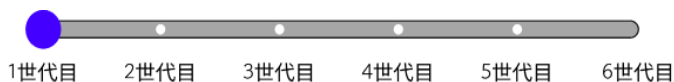
d : お見合い範囲データ
 ω : データの重み

集合知の経験年数選択

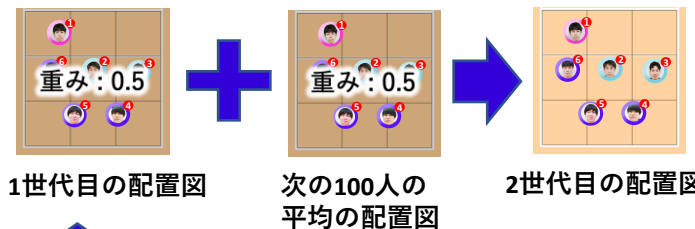


配置画面で選んだ経験年数バー別に分け、それぞれで集合知を取ります。

集合知の世代別表示



世代選択バーを動かして、DBから配置データを登録した順で集合知として集めます。この例では100人を1世代とします。



指定した世代になるまで繰り返します。

選手の評価

選手は各経験年数の集合知の配置を見て、その配置通りにつきます。実際に配置してみても、評価してもらいます。

選手全員に集合知の配置図の気に入り具合を、自分の守備範囲が明確かどうかの観点で聞き、選手6人の平均をとって表示します。

類似品との比較・独創的な点

類似する特許・システム

特開2016-93402(P2016-93402A)

実際の過去から現在までの野球の試合映像を利用して野手の守備範囲を求めるシステム。

何試合もの映像が必要になるため現状の守備範囲とズレが生じる。



数秒程の短い動画から他者の意見を取り入れることで選手間のお見合い範囲を求めて表示できる。

現状の守備範囲を知ることができる！

特開2004-321601(P2004-321601A)

ゲーム空間上で勢力分布を表示しサッカーの優勢範囲やスペースのできた領域を表示するシステム。

利用者一人では学べない。



自分で作った配置はもちろん、観客が作った配置を集合知として見られる。

オンライン上でみんなで学べる！

バレーボール作戦ボードパワー

バレーボールコート上の好きな位置にコマを作成し配置できるシステム。

作戦立案者の経験年数を情報として加味できない。



システム利用者の経験年数を登録したあとでコート上にコマを配置する。

他者の経験年数ごとに配置を比較できる！

My Formation

予め定められたサッカーのフォーメーションを軸に配置し戦術を考えるシステム。

どこが守備範囲の欠点になるかわからない。



お見合い範囲を可視化し、より良い戦術を考えることができる。

守備範囲の欠点部分を可視化できる！

独創的な点

- ・お見合い範囲を表示しながら選手配置をすることができる。
- ・経験年数ごとにどのように配置しているかを知ることができる。
- ・多くの人の意見を集めて比較することで最強の配置を考えられる。

開発スケジュール・開発環境

	4月	5月	6月	7月	8月	9月	10月
要求分析	アイデア出し 機能決定						
システム設計		データベース設計 レイアウト設計					
Webページ		レイアウト 検討	レイアウト 決定				
学習データ用 動画撮影	テスト動画の 撮影	データ収集用動画の撮影					
機械学習用データ	ダミーデータ 作成	学習データ 取得	学習データの見直し・調整				
お見合い範囲 計算用データ	ダミーデータ 作成	テストデータ 取得	テストデータの見直し・調整				
機械学習	数値分析・APIの作成						
実証実験					実証実験・システムの調整		

■開発環境・実行環境

開発OS：Windows10

開発言語：HTML、CSS、JavaScript、Python

IDE：Visual Studio Code

サーバー：Apache、MySQL