

動きの緩急をコントローラとするインタラクティブゲーム「風林火山」

藤井 叙人, 片寄 晴弘
関西学院大学理工学部

本論文では、ビデオフレーム間差分によってプレイヤーの動きの緩急と方向をとらえ、その情報に基づいて動作する複合現実空間プレイヤ投影型のスパイ風ゲーム「風林火山」の制作事例報告を行う。「風林火山」はプレイヤーが敵に気付かれないようにゆっくりと敵に近づいた時に攻撃が可能で、逆に、大きく（速く）動いて、気付かれてしまった場合、敵に追尾され、攻撃を受けるというシンプルなコンセプトを実現したゲームであるが、実際に動き回ってゲームに参加することによる緊張感・臨場感はこのほか高い。「風林火山」のデザインコンセプト、プロトタイプシステム、今後の課題について述べる。

Interactive game “Fu-rin-ka-zan” with control system of motional speed

Nobuto FUJII, Haruhiro KATAYOSE
Kwansei Gakuin University, the School of Science & Technology

In this paper, we propose an interactive game “Fu-rin-ka-zan” in which players are projected like a secret agents in augmented reality. This game, as controllers, adopts motional speed and direction obtained from differences between video-frames. The player can attack on enemy if she/he could approach enemies within a slow speed that enemy could not notice. If the enemies notice player’s motion, the enemies start to chase the player. “Fu-rin-ka-zan”, the design of which is based on a simple concept, provides the players higher tension due to presence that the players can move around in the real field. In this paper, we describe the design concept, a prototypical system, and discuss the problems for the next implementation of “Fu-rin-ka-zan”.

1. はじめに

ゲームの定義・楽しみは、1つに限定されるものではないが、プレイヤーがある特定のキャラクタまたはモノになりきり、現実とは違った独立した空間の中で、自分の役割に達成感を感じ、楽しみを見出すという点は、その本質を言い表していると思われる。実際、市販ゲームのデザインにおいては、プレイヤーがキャラクタになりきり、よりリアルな感情移入を促すことを目的として、数多くの工夫がなされてきた。

従来のゲームのコントローラとしては、両手に収まるサイズで、方向キーと数種のボタンによって操作するインタフェースが利用されてきた。3Dスティックやヴァイブレーション

機能が搭載されているもの、また、銃型、ハンドル型のコントローラも開発されている。しかし、グラフィックやサウンドに関しては現実と見間違ふほどに技術が進歩した現在のゲームにおいて、従来のコントローラでは没入感の視点からみると、どうしても、もの足りないという側面が指摘されている。この観点から、ARを応用したゲーム[1,2,3]やインタフェースに人形アバタを活用したもの[4]などが提案されてきた。Magic Book[1]は、加藤によって提案されたARToolkitを用いた飛び出す絵本である。Magic Bookの他にも、ARToolkitの応用例は数多い。ビデオゲームとして成立しているものをAR環境によって再構成したものとしては、長谷川らの格闘ゲ

ーム[2], 今西らのトレーディングカードゲームの実装[3]などの研究例がある.

我々は, 手のみを使った操作ではなく, からだの動きをコントローラとして利用するエンタテインメントの実装を課題としている. 本論文では, 具体的なコンテンツ例として, 「敵にゆっくり近づいた時に攻撃が可能で, 逆に, 動きが大きい時は敵に追跡される」というコンセプトを実現したインタラクティブゲーム「風林火山」の制作実施状況について報告する. ここでは, プレイヤ自身をゲーム内に投影することで, プレイヤはキャラクターになりきることができ, 同時に, からだ全体を動かすことによりプレイヤに能動的な没入感を与える効果を期待している. 特に, 敵に気づかれないようゆっくりと動く時の緊張感を, より際立たせるようなシステム構成になっている.

2. 「風林火山」概要

2.1 デザインコンセプト

「風林火山」は, 敵に気づかれないようにゆっくりと敵に近づき, 敵を殴る, 蹴る等の動作を行った時に, プレイヤは攻撃が可能であり, また, 逆に, 大きく(速く)動いて, 敵に気づかれてしまった場合, プレイヤは敵に追尾され攻撃を受ける, というシンプルなコンセプトを実現したスパイ風のゲームである. プレイヤの動きの緩急と方向をとらえるにあたっては, ビデオのフレーム間差分を利用する.

2.1 実施例

「風林火山」の実施例を図1に示す. ビデオカメラを斜め上に設置し, ビデオカメラの視界内全体をプレイヤが移動できるフィールドとする. ビデオカメラの映像は, PCで複合現実空間を生成し, プロジェクターによってプレイヤの正面に投影する. この際, プレイヤの影が投影された画面内に映らないように, プロジェクターは下方に設置し, 2枚のミラーを用いて投影することで, プレイヤが自由自在に動き回ることのできるフィールドを実現した.

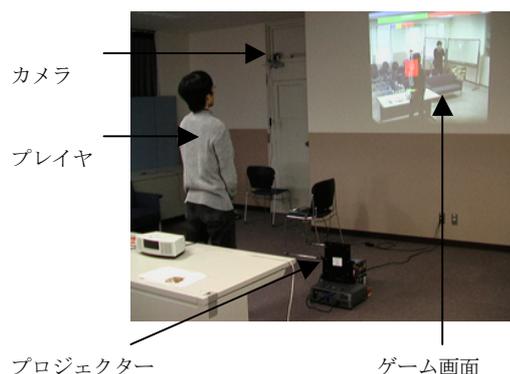
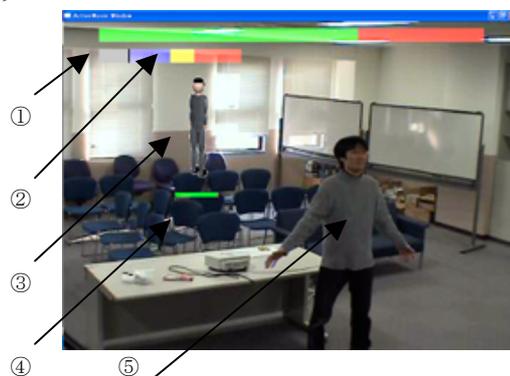


図1. ゲーム環境全体図

ゲーム画面としては, ビデオカメラから得た映像に, 敵グラフィックや動きの大きさを示すバー, プレイヤと敵のHPバーをスーパーインポーズしたものを表示する(図2). プレイヤは映像や効果音, 音楽から敵の位置や敵状態を認識し, 敵を攻撃する, 敵から逃げる, 敵から隠れる, といったアクションを行う.



①プレイヤーのHP ②動きの緩急
③敵 ④敵のHP ⑤プレイヤー

図2. ゲーム画面

3. システム概要

3.1 動きの抽出

ビデオカメラの映像から, リアルタイム動画像処理により, 現在フレームとその1フレーム前との差分を求める. 差分の判定は, 画素のRGB値の変化量がある閾値以上の時に, その画素において差が生じた, と認識する手法を用いた. 1フレーム間で動いた画素数を計算し, これを変数 *move* に加算することに

よって、動きの緩急を積分系で表す。変数 `move` は、敵のモードに応じた一定値を1フレームごとに減算するようにする。つまり、1フレームの間に、減算される一定値以上の動きを行った場合、変数 `move` は増加することになる。

敵への与ダメージ、敵からの被ダメージも差分を用いて算出する。1フレーム前との差分と敵画像が重なっている部分の画素数を求め、その値をダメージとする。与ダメージ、被ダメージの切り替えは敵のモードによって行う。敵のHPが0になった場合は、ランダムに新規の敵を出現させる。プレイヤーのHPが0になった場合はゲームオーバーとする。

3.2 プレイヤの状態

動きの緩急(図2.②)に応じてプレイヤーの状態は以下の4つに変化する。閾値は変数 `move` の値により定める(図3)。

- ・透明状態
閾値V以下が一定時間以上継続した場合、プレイヤーは徐々に透明になる。
- ・青状態
閾値V以上Y以下の場合、プレイヤーに青色の残像エフェクトが掛かる。
- ・黄状態
閾値Y以上R以下の場合、プレイヤーに黄色の残像エフェクトが掛かる。敵は警戒モードに移行する。
- ・赤状態
閾値R以上の場合、プレイヤーに赤色の残像エフェクトが掛かる。敵は追跡モードに移行する。

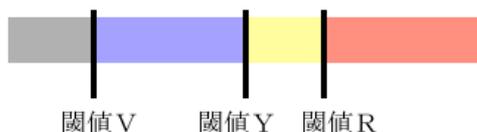


図3. 動きの緩急(変数 `move`)の閾値

透明状態を実現するために、あらかじめ背景フレームとして初期フレームを保持してお

き、現在フレームと背景フレームを徐々に切り替える手法を用いた。

また、動きの緩急とは別に、敵の向いている方向と逆の場所にプレイヤーがいる場合は白状態になる。

- ・白状態
プレイヤーに白色の残像エフェクトが掛かる。変数 `move` の値が増加しない。敵へのダメージが倍に増加する。

白状態の判定は、1フレーム間で変化した画素の重心と敵の座標との方向性を求め、敵の向きと求めた方向が逆の場合は、敵の後ろ側にいると認識する手法で実現した。

3.3 敵のモード

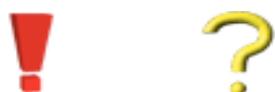
敵のモードはプレイヤーの状態に依存して、3つのモードに切り替わる。

- ・索敵モード
通常時のモードであり、与ダメージ判定となる。敵は一定方向に動くのみである。
- ・警戒モード
プレイヤーが黄状態になった時に移行するモードであり、与ダメージ判定となる。敵は最初に一度追跡計算により速度を更新し、プレイヤー付近を重点的に巡回する。プレイヤーは動きを小さくすることで変数 `move` を減少させて透明状態になるか、透明状態を一定時間維持することにより、敵は索敵モードへ戻る。
- ・追跡モード
プレイヤーが赤状態になった時に移行するモードであり、被ダメージ判定に切り替わる。敵は一定フレームごとに追跡計算を行うことで、速度を更新し動きの大きい部分を追跡する。プレイヤーは敵から逃げるか、動きを小さくすることで変数 `move` を減少させ、透明状態になれば、敵は警戒モードに戻る。

警戒モード、追跡モードでは閾値V、閾値

Yを索敵モードより小さくすることで、黄色状態になる範囲を広げる。さらに、変数 moveの減少量を半減し、敵が極めて注意深い状態であることを表現した。

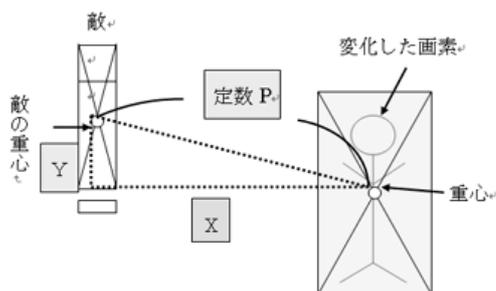
敵のモードの切り替え時には、敵のグラフィックに「!」「?」を表示することで視覚的に、また、効果音と音楽の変更によって聴覚的にプレイヤーへのフィードバックをおこなう(図4)。



追跡モード 警戒モード
図4. 「!」と「?」のグラフィック

3.4 追跡計算

1フレーム間で変化した画素の重心と、敵画像の重心を結ぶ直線を斜辺とした直角三角形を考え、斜辺を定数Pとした時のX、Yを求めることで、X軸方向の速度、Y軸方向の速度を算出する(図5)。敵が警戒状態になった時、追跡状態の時にこの計算を行なうことで、敵はプレイヤーを追跡することが可能となる。



$$X = P * (X / \text{Sqrt}(X^2 + Y^2)) \rightarrow \text{X軸方向の速度}$$

$$Y = P * (Y / \text{Sqrt}(X^2 + Y^2)) \rightarrow \text{Y軸方向の速度}$$

図5. 追跡計算のイメージ図

4. 考察とまとめ

ビデオカメラを用い、動きの緩急をコントローラとしたことで、プレイヤーは、フィールド上を実際に動き回ってゲームに参加することが可能となった。その結果、敵から見つか

らないように努力する緊張感や、敵に忍び寄り倒したときの達成感を、よりリアルに感じることができるようになった。また、実際に体を動かすことによる臨場感を得られるようになった。今後は、これらゲームの印象に関して、定量的な分析を行っていく予定である。具体的には、ゲーム好きのユーザと、そうでないユーザに分けての、緊張感、達成感、臨場感の内観に関する数量的分析、脳活動計測に基づく分析を実施する予定である。さらに、細部のデザインの改良を実施し、システムの完成度を高めていきたい。

参考文献

- [1] Campbell.S, Billinghamst.M, Kato.H, Poupyrev.I : “Magic Book”, Exploring Transitions in Collaborative AR Interfaces ACM SIGGRAPH 2000, (2000)
- [2] Shoichi Hasegawa, Toshiaki Ishikawa, Naoki Hashimoto : “Human Scale Haptic Interaction with a Reactive Virtual Human in a Realtime Physics Simulator”, Proc. ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE 2005), FP3-6(a66), (2005)
- [3] Haruhiro Katayose, Kazuhiko Imanishi : “ARMS: A Trading Card Game using AR Technology”, Proc. ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE 2005), Poster1(a45), (2005)
- [4] Noriyoshi Shimizu, Naoya Koizumi, Maki Sugimoto, Hideaki Nii, Dairoku Sekiguchi, Masahiko Inami : “Teddy-bear Based Robotic User Interface”, Proc. ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE 2005), FP2-3(a36), (2005)