

タブレット端末向け VR 迷路ゲームの作成 Creating a tablet terminal VR maze game

○吉田和¹, 山口健², 吉川浩³Nagomi Yoshida^{1,*}, Takeshi Yamaguchi², Hiroshi Yoshikawa³

Abstract: In this study, we create a tablet terminal VR maze game. To create a complex maze, we compare put bar down method with hole digging method of automatic maze generation algorithm. This game uses Unity of multi-platform game engine, which improve the gameplay and graphics. Enemy characters and action of the player are also implemented. In addition, the operation on the tablet terminal is enabled.

1. まえがき

本報告では, 迷路のグラフィックとゲーム性を向上させるためゲームエンジンである Unity^[1]を用いて迷路を作成する. 迷路の自動生成方法を比較することで迷路の複雑性を向上し, プレイヤーのアクションや敵キャラクター等の配置を行う. また, タブレット端末での操作を可能にする.

2. アプリケーションの概要

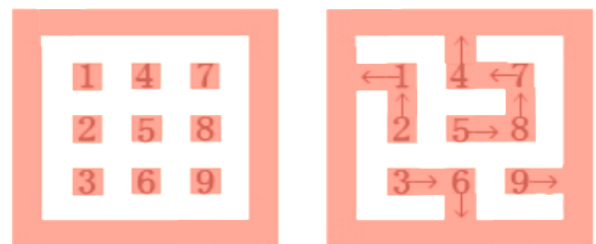
本報告で作成する迷路は, 3D 迷路の中で 3D キャラクターを操作し, ゴールまで到達するものである. ゴールに到達した場合クリア画面を表示する. ゲーム性を向上させるためにプレイヤーにジャンプや攻撃などのアクションを加え, 落とし穴や敵キャラクター等の障害物を設置する. また, 迷路自動生成アルゴリズムである棒倒し法と穴掘り法を比較する.

3. 迷路の自動生成

本報告では, 迷路自動生成アルゴリズムである棒倒し法^[2]と穴掘り法^[2]で生成した迷路を比較する. 迷路の大きさは 31×31 とする.

Figure 1 に棒倒し法による自動生成の概念図を示す. まず, Figure 1(a)のように外側の壁と, 内側に 1 マス間隔ずつ壁を生成する. 次に, (b)のように内側の壁を左側(1, 2, 3) から 1 つずつ順番に上下左右何れかの方向へ伸ばす. すでに壁がある場所には壁を伸ばさないようにする. このようにすることで, 迷路が自動生成される. しかし棒倒し法は, 基準点を作るため袋小路が短く, 迷路が似たような形になりやすいため複雑性に欠けている.

Figure 2 に穴掘り法の概念図を示す. 穴掘り法は迷路上の道を生成するための基準点が決まってい



(a) Creating a reference point

(b) Creating a wall

Figure 1. Put bar down method.



(a)Creating a passage

(b)No passage

(c) Failure to create

Figure 2. Hole digging method.

ないため, 道のランダム性が高くなる. まず, 迷路全体を壁にする. 次に Figure 2(a)のように, 上下左右のいずれかの方向の 2 マス先を見て, そこが通路でなければ道を伸ばす. Figure 2(b)の●の位置のように上下左右の 2 マス先がすべて通路になってしまった場合は, ▲の位置のようにすでにある通路からランダムに場所を選び, 道を伸ばしていく. ランダムに選ばれた場所が(c)のようにどの方向にも道を伸ばすことが出来ない場合は, 再び別の場所を探す. この作業を 1000 回繰り返し, その後スタート地点が選択されると終了する.

4. ゲーム性の向上

ゲーム性を向上させるために迷路中に落とし穴や敵キャラクターを設置する. 落とし穴は中央部に衝

1 : 日大理工・学部・子情 2 : 日大理工・教員・電子 3 : 日大理工・教員・情報

突判定のついた透明な立方体があり、立方体に触れるとゲームオーバー画面へ移行する。敵キャラクターは、初めFigure 3(a)のように待機状態になっている。しかし、敵を中心に設定された範囲内にプレイヤーが近づくと、Figure 3(b)のように攻撃状態へ切り替わり、敵キャラクターはプレイヤーを追いかける。敵キャラクターは3の体力を持っており、プレイヤーが3回攻撃動作を行うと、気絶状態になり動かなくなる。

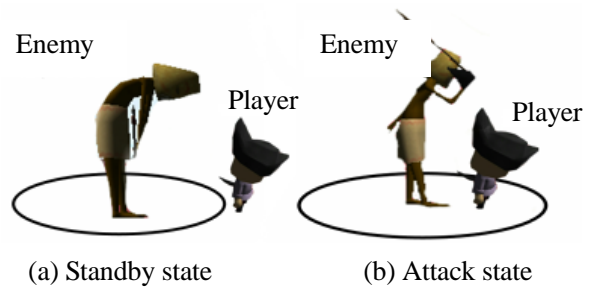
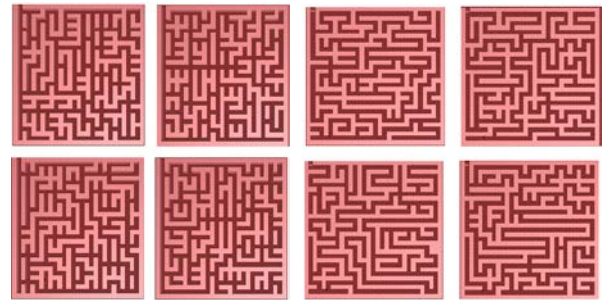


Figure 3. State of the enemy.

本報告では、マルチプラットフォーム対応のゲームエンジンであるUnityを用いてプレイヤーの操作をタブレット端末の操作に対応させる。

5. 結果

Figure 4(a)に棒倒し法で作成した迷路, (b) に穴掘り法で作成した迷路を示す。迷路の大きさは、31×31とした。迷路の複雑性を比較するために迷路を4回生成し、迷路の交差点数を比較する。迷路の交差点数が少ないほど、袋小路が多く入り組んだ迷路になる。交差点数の比較Table .1 に示す。棒倒し法での交差点数の平均は56個になり、穴掘り法での平均は19個になった。これにより、穴掘り法で複雑性の高い迷路を作成することができた。



(a) Put bar down method (b) Hole digging method

Figure 4. Comparison of put bar down method and hole digging method .

Table .1 Comparison of the number of intersection

	1	2	3	4	Average
Put bar down method	51	68	49	58	56
Hole digging method	21	16	19	22	19

落とし穴の様子をFigure 5に示す。プレイヤーは、ジャンプ動作で落とし穴を飛び越えることができる。プレイヤーが落とし穴内に入るとゲームオーバー画面に移行する。



Figure 5. A pitfall.

作成した敵キャラクターの様子をFigure 6に示す。初期状態の敵キャラクターは待機状態であるが、プレイヤーが攻撃範囲内に入ると敵キャラクターは攻撃動作に入りプレイヤーを追う。プレイヤーの攻撃を敵キャラクターに3 回当てると、動かなくなる。



Figure 6. Encountering an enemy.

タブレット端末での操作の様子をFigure 7に示す。プレイヤーはタブレット端末のタップやジャイロ機能を用いて操作を行う。



Figure 7. Operation on a tablet terminal.

6. むすび

本報告では、タブレット端末向け迷路ゲームの作成を行った。迷路の自動生成方法を穴掘り法にし、より複雑な迷路とした。プレイヤーにジャンプや攻撃等の動作を持たせ、落とし穴や敵キャラクターを作成したことによりゲーム性の向上させた。また、タブレット端末での操作を可能にした。

7. 参考文献

[1] Unity, <http://japan.unity3d.com/>
 [2] 迷路自動生成アルゴリズム
<http://www5d.biglobe.ne.jp/257estssk/maze/make.htm>